



普通高等教育“十一五”国家级规划教材



电子信息与电气学科规划教材·自动化专业

微型计算机控制技术

徐安 陈耀 李锐华 郭其一 编著 白英彩 审



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY <http://www.phei.com.cn>

普通高等教育“十一五”国家级规划教材
电子信息与电气学科规划教材·自动化专业

微型计算机控制技术

徐 安 陈 耀 李锐华 郭其一 编著

白英彩 审

電子工業出版社·

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书围绕“微机控制算法及其实现”这一主题，系统而有重点地讲述了运动控制、常用控制、现代控制和智能控制中的各种主要算法的思路精髓和微机实现这些算法的要领，并从实际应用的角度出发，介绍了微机控制系统与信号的数学描述、常用数据结构和数据处理、嵌入式操作系统基础、系统设计及电磁兼容技术。根据信息与控制日益交融的发展趋势，介绍了微机控制网络与现场总线。

本书力求从电类专业“软（件）硬（件）兼施”、“强（电）弱（电）结合”的特色出发，对微型计算机控制技术的学习和应用提供切实的指导和帮助。本书可作为高等院校有关课程的教材，也可作为电子技术人员的参考书。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有，侵权必究。

图书在版编目 (CIP) 数据

微型计算机控制技术 / 徐安等编著. —北京：电子工业出版社，2008.10

电子信息与电气学科规划教材·自动化专业

ISBN 978-7-121-07394-6

I. 微… II. 徐… III. 微型计算机—计算机控制—高等学校—教材 IV. TP273

中国版本图书馆 CIP 数据核字（2008）第 142029 号

策划编辑：凌 毅

责任编辑：余 义

印 刷：

装 订：

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编：100036

开 本：787×1092 1/16 印张：22.25 字数：570 千字

印 次：2008 年 10 月第 1 次印刷

印 数：4000 册 定价：33.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：(010)88254888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：(010)88258888。

前 言

目前,绝大多数自动控制都是使用计算机来实现的。微型计算机控制技术的发展,使得以微型计算机为控制器核心的嵌入式测控装置与系统广泛渗透到国民经济的各行各业中,已经无时无刻、无处不在地影响着每个现代人的生活。

在我国高等工科院校中,普遍开设了微型计算机控制技术及其相关课程。本书围绕“微机控制算法及其实现”这一主题,系统而有重点地讲述了运动控制、常用控制、现代控制和智能控制中的各种主要算法的思路精髓和微机实现这些算法的要领,并从实际应用的角度出发,介绍了微机控制系统与信号的数学描述、常用数据结构和数据处理、嵌入式操作系统基础、系统设计及电磁兼容技术。根据信息与控制日益交融的发展趋势,介绍了微机控制网络与现场总线。

微型计算机控制技术内容十分丰富,每一个专题的详细阐述都需要一本或几本专著。本书注意了与“微控制器原理与应用”等相关课程的分工协调、统筹兼顾,力求从电类专业“软(件)硬(件)兼施”、“强(电)弱(电)结合”的特色出发,对微型计算机控制技术的学习和应用提供切实的指导和帮助。本书可作为高等院校有关课程的教材,也可作为电子技术人员的参考书。

实验和实践对微型计算机控制技术的学习和应用极为重要。本书在第11章中,以DP-01多MCU实验系统为平台,以多种控制算法、数据传输方式与总线在智能家居中的应用为主题,给出了微型计算机控制技术的设计性、综合性实验大纲,引导和鼓励學生发挥想象力和创造性。

多年来,于海生、周德泽、王晓明、桑楠、谢剑英、陈炳和、高金源、王慧、齐蓉、何玉彬、蔡德聪、阳宪惠、何克忠、袁南儿、蔡自兴、张乃尧、白英彩、吴启迪、窦振中、涂时亮、王常力、陈汝全、金以慧、王锦标、严隽薇、周立功、潘新民、胡寿松、吴坚等教授在微型计算机控制技术的教育园地里辛勤耕耘,本书多处引用了他们的研究成果,谨在此表示衷心的感谢。

广州周立功单片机发展有限公司上海分公司经理曾成奇、微芯科技咨询(上海)有限公司经理刘晖、飞思卡尔半导体(中国)有限公司上海分公司经理张明峰、高拓微电子技术有限公司经理魏敏和吴健等为本书提供了重要帮助,谨在此表示诚挚的感谢。

担任本书主审的上海交通大学白英彩教授认真审阅了全书,提出了指导性的建议和中肯的意见;研究生吴霄、张振富、顾鸣、郁海华、郭凯杰和丁才云等为程序设计和图文制作输入付出了辛勤的劳动,在此谨致以深切的谢意。

本书由徐安组织编写,李锐华、陈耀、郭其一分别参加了第3,6章和第7,8,10章的编写。本书是普通高等教育“十一五”国家级规划教材,得到了同济大学教材、学术著作出版基金委员会的资助。

本书提供配套的电子课件,可登录电子工业出版社的华信资源教育网:www.huaxin.edu.cn或www.hxedu.com.cn,注册后免费下载。

限于编者水平,书中错误和不妥之处在所难免,恳请读者批评指正。联系地址:xuanb@online.sh.cn。

编 者

目 录

第 1 章 概论	(1)
1.1 控制用微型计算机	(1)
1.1.1 控制用微型计算机的类型	(1)
1.1.2 控制用微型计算机的选用	(3)
1.2 微型计算机控制系统	(3)
1.2.1 微机控制系统的组成	(3)
1.2.2 微机控制系统的工作过程与方式	(4)
1.2.3 微机控制系统的功能与特点	(5)
1.2.4 微机控制系统的结构	(7)
1.3 微机控制技术的发展与作用	(10)
1.3.1 控制理论与控制技术的发展	(10)
1.3.2 微机控制技术的发展	(11)
1.3.3 微机控制技术的作用	(11)
1.3.4 微机控制技术的前景	(11)
习题与思考题	(12)
第 2 章 常用数据结构和数据处理	(13)
2.1 常用数据结构	(13)
2.1.1 数据结构的基本概念	(13)
2.1.2 控制系统的特殊数据结构	(20)
2.2 采样与数据处理	(22)
2.2.1 采样周期的选择	(22)
2.2.2 常用数据处理	(24)
2.3 数字滤波	(26)
2.3.1 常用数字滤波方法	(27)
2.3.2 数字滤波方法的选择	(29)
习题与思考题	(29)
第 3 章 微机控制系统与信号的数学描述	(30)
3.1 微机控制系统中的信号变换	(30)
3.1.1 信号的采样与保持	(31)
3.1.2 采样定理	(31)
3.1.3 离散时间信号	(32)
3.2 线性定常离散系统的描述与差分方程	(33)
3.2.1 线性定常离散系统的描述	(33)

3.2.2	线性常系数差分方程	(34)
3.2.3	应用迭代法求解差分方程	(34)
3.3	Z 变换及其应用	(35)
3.3.1	Z 变换定义	(35)
3.3.2	Z 变换方法	(36)
3.3.3	Z 逆变换	(42)
3.3.4	应用 Z 变换求解差分方程	(45)
3.4	脉冲传递函数	(46)
3.4.1	脉冲传递函数定义	(46)
3.4.2	零阶保持器及其脉冲传递函数	(47)
3.4.3	开环系统脉冲传递函数	(49)
3.4.4	闭环系统脉冲传递函数	(50)
	习题与思考题	(53)
第 4 章	运动控制技术	(54)
4.1	顺序控制	(54)
4.1.1	顺序控制原理	(54)
4.1.2	顺序控制系统特性	(55)
4.1.3	专用顺序控制器设计	(55)
4.1.4	通用顺序控制器的选用	(56)
4.2	程序控制	(56)
4.2.1	程序控制原理	(56)
4.2.2	逐点比较法直线插补	(57)
4.2.3	逐点比较法圆弧插补	(59)
4.2.4	运动控制芯片	(62)
4.3	可编程控制器	(63)
4.3.1	PLC 的主要功能与特点	(63)
4.3.2	PLC 的结构与工作原理	(64)
4.3.3	PLC 常用编程语言	(67)
4.3.4	PLC 的选用	(70)
4.4	直流电动机控制	(71)
4.4.1	直流电动机调速原理	(72)
4.4.2	直流电动机不可逆 PWM 系统	(74)
4.4.3	直流电动机可逆 PWM 系统	(75)
4.4.4	专用运动控制处理器	(79)
4.5	交流异步电动机控制	(81)
4.5.1	交流异步电动机变频调速原理	(81)
4.5.2	变频调速的机械特性及其补偿	(86)
4.5.3	SPWM 波发生器及其应用	(87)
4.5.4	矢量控制与直接转矩控制	(88)

4.6	步进电动机控制	(91)
4.6.1	反应式步进电动机的结构和工作原理	(91)
4.6.2	步进电动机的工作方式	(93)
4.6.3	步进电动机特性	(95)
4.6.4	步进电动机的驱动方式	(98)
4.6.5	步进电动机的控制	(100)
4.7	无刷直流电动机控制	(102)
4.7.1	无刷直流电动机的原理与结构	(102)
4.7.2	无刷直流电动机的绕组连接方式	(104)
4.7.3	无刷直流电动机的控制	(108)
	习题与思考题	(111)
第 5 章	常用控制技术	(112)
5.1	数字 PID 控制	(112)
5.1.1	PID 控制	(112)
5.1.2	PID 算法的计算机实现	(114)
5.1.3	PID 算法的改进	(115)
5.1.4	PID 控制的参数整定	(120)
5.2	纯滞后补偿控制	(123)
5.2.1	大林 (Dahlin) 算法	(123)
5.2.2	史密斯 (Smith) 预估控制	(126)
5.3	最少拍控制	(127)
5.3.1	不同控制对象的最少拍控制设计	(127)
5.3.2	不同输入函数的最少拍控制设计	(129)
5.3.3	最少拍控制设计的局限性	(131)
5.3.4	最少拍无纹波控制的设计	(132)
5.4	串级控制	(133)
5.4.1	串级控制系统结构和工作原理	(134)
5.4.2	串级控制系统的特点	(134)
5.4.3	串级控制系统的设计	(136)
5.4.4	串级控制系统的应用	(137)
5.5	前馈控制	(138)
5.5.1	前馈控制原理与结构	(138)
5.5.2	前馈-反馈控制	(138)
5.5.3	前馈-串级控制	(139)
5.5.4	前馈控制的设计原则与参数整定	(139)
5.6	线性数字控制器的计算机实现	(140)
5.6.1	线性数字控制器的一般形式	(140)
5.6.2	数字控制算法的计算机实现	(141)
	习题与思考题	(142)

第 6 章	现代控制技术	(144)
6.1	线性离散系统的状态空间描述	(144)
6.1.1	MIMO 线性离散系统的状态空间描述	(144)
6.1.2	线性离散系统状态方程的求解	(157)
6.1.3	线性离散系统的 Z 传递矩阵和 Z 特征方程	(159)
6.1.4	微型计算机控制系统的离散状态空间表达式	(161)
6.2	离散系统的能控性和能观测性	(164)
6.2.1	离散系统的能控性	(164)
6.2.2	离散系统的能观测性	(167)
6.3	离散系统的状态空间设计	(169)
6.3.1	MIMO 线性离散系统的状态空间设计方法	(169)
6.3.2	MIMO 离散系统设计实例	(170)
	习题与思考题	(177)
第 7 章	智能控制技术	(179)
7.1	模糊控制	(179)
7.1.1	模糊数学基础	(179)
7.1.2	模糊关系与模糊推理	(185)
7.1.3	模糊控制原理	(191)
7.1.4	模糊控制设计实例	(195)
7.2	专家控制	(200)
7.2.1	专家系统	(200)
7.2.2	专家控制系统	(203)
7.2.3	专家控制系统的设计	(206)
7.3	人工神经网络控制	(208)
7.3.1	人工神经网络的基本模型	(209)
7.3.2	人工神经网络学习规则	(219)
7.3.3	人工神经网络控制的结构方案	(221)
7.3.4	人工神经网络控制的特点	(225)
7.4	多种控制策略的渗透和结合	(226)
7.4.1	模糊神经网络 (FNN) 控制	(226)
7.4.2	神经网络专家系统	(228)
7.5	基于进化和群体智能的控制算法	(230)
7.5.1	遗传算法	(230)
7.5.2	蚁群算法	(232)
	习题与思考题	(235)
第 8 章	嵌入式操作系统基础	(236)
8.1	嵌入式系统	(236)
8.1.1	嵌入式系统的概念	(236)
8.1.2	嵌入式系统的体系结构	(236)

8.1.3	嵌入式系统的特点	(237)
8.1.4	嵌入式系统的类型	(237)
8.2	嵌入式操作系统	(238)
8.2.1	嵌入式操作系统的基本概念	(238)
8.2.2	嵌入式操作系统的实现技术	(239)
8.2.3	嵌入式操作系统的性能指标	(242)
8.2.4	嵌入式操作系统的分类	(243)
8.2.5	使用嵌入式操作系统的优缺点	(244)
8.3	典型的嵌入式操作系统	(245)
8.3.1	嵌入式 Windows	(245)
8.3.2	嵌入式 Linux	(250)
8.3.3	VxWorks	(255)
8.3.4	Nucleus PLUS	(256)
8.3.5	RTEMS	(258)
8.3.6	μ C/OS-II	(261)
8.4	嵌入式操作系统应用	(262)
8.4.1	嵌入式系统的选择	(262)
8.4.2	嵌入式系统的移植	(264)
8.4.3	嵌入式软件的发展趋势	(267)
	习题与思考题	(268)
第 9 章	网络与现场总线	(269)
9.1	数字通信与网络	(269)
9.1.1	总线的基本概念	(269)
9.1.2	数字信号传输与通信系统	(270)
9.1.3	计算机局域网	(272)
9.1.4	网络协议及其层次结构	(282)
9.2	分布式控制系统 (DCS)	(284)
9.2.1	DCS 结构与特点	(284)
9.2.2	直接测控级	(286)
9.2.3	过程优化级	(288)
9.2.4	生产管理、经营决策和 CIMS	(290)
9.3	现场总线控制系统 (FCS)	(292)
9.3.1	现场总线技术特点与优点	(292)
9.3.2	现场总线的调度策略	(293)
9.3.3	现场总线的 OSI 模型	(294)
9.4	典型的现场总线	(294)
9.4.1	基金会现场总线 (FF)	(294)
9.4.2	LonWorks	(297)
9.4.3	Profibus	(300)

9.4.4	CAN	(303)
9.4.5	现场总线的困境	(304)
9.4.6	工业以太网	(305)
9.4.7	信号线与电源线合用电缆的研究	(309)
习题与思考题		(310)
第 10 章 微机控制系统设计		(311)
10.1	微机控制系统设计与实现	(311)
10.1.1	微机控制系统设计原则	(311)
10.1.2	系统的设计与实现过程	(312)
10.2	输入输出通道	(313)
10.2.1	开关量信号的调整	(313)
10.2.2	模拟量信号的调整	(314)
10.2.3	输入输出模板设计	(316)
10.3	电磁干扰的抑制、屏蔽与接地	(318)
10.3.1	共模与串模干扰的抑制	(318)
10.3.2	长线传输	(321)
10.3.3	信号线的选择与布线	(322)
10.3.4	屏蔽与接地	(324)
10.3.5	电源	(325)
10.4	人机界面与监控组态软件	(326)
10.4.1	人机界面	(326)
10.4.2	监控组态软件的特点	(328)
10.4.3	组态软件的数据流与典型构成	(329)
习题与思考题		(334)
第 11 章 微机控制技术实验		(335)
11.1	实验大纲	(335)
11.1.1	顺序控制及其在门禁系统中的应用	(335)
11.1.2	数字滤波及其在灯光子系统中的应用	(336)
11.1.3	运动控制及其在输送子系统中的应用	(337)
11.1.4	PID 控制及其在温度子系统中的应用	(338)
11.1.5	模糊控制及其在智能终端子系统中的应用	(339)
11.1.6	多种数据传输方式与总线在智能家居中的应用	(341)
11.1.7	网络与各子系统的整合在智能家居中的应用	(341)
11.2	DP—01 多 MCU 实验平台	(342)
11.2.1	DP—01 多 MCU 实验平台布局	(342)
11.2.2	DP—01 多 MCU 实验平台功能模块	(343)
参考文献		(344)

第 1 章 概 论

自动控制的主要功能是：没有人的直接参与，利用外加的设备或者装置，使被控对象（机器、设备或生产过程）的被控量（工作状态或参数）自动地按照预定的规律运行。在现代科学技术的众多领域中，自动控制技术起着越来越重要的作用。

目前，几乎所有的控制系统都是基于计算机来实现的。微型计算机控制技术的发展，使得以微型计算机为控制器核心的嵌入式测控装置与系统广泛渗透到各个领域中，已经无时无刻、无处不在地影响着每个现代人的生活。

1.1 控制用微型计算机

自从 20 世纪 70 年代初 Intel 公司生产出第一个微处理器 4004 以来，随着半导体技术的进步，微型计算机得到了飞速的发展。已从 4 位机、8 位机、16 位机、32 位机，发展到目前的 64 位机。就微型机种类而言，不但有早期的 8086/8088，80286，80386，80486，以及相继问世的奔腾系列等功能齐全的高性能微型计算机，而且还出现了许多小巧灵活、从 8 位到 16 位的微控制器，由于各种复杂运算的需要而出现的 DSP，以及 32 位 ARM 处理器，这些都进一步拓展和细化了微机应用的广度和深度。

从应用角度出发，微型计算机可以分为面向事务处理的通用型微机和面向实时监测控制的控制用微机。除了微型计算机的共同特性外，控制用微机在设计思想、系统结构、硬软件组成、开发工具、使用方法等方面还有许多特点，例如：

- 高可靠性；
- 软件和操作系统的实时性；
- 丰富的输入输出（I/O）接口；
- 多样的人机接口。

控制用微机的组成如图 1.1 所示，根据实际应用的需要，控制用微机所含的软件、硬件各不相同。

1.1.1 控制用微型计算机的类型

1. 单片微型计算机

单片微机（包括 DSP），是工业控制和智能化系统中应用最多的模式。这种模式的最大特点是设计者可根据自己的实际需要进行开发、设计，在单片机的基础上扩展一些接口，并开发一些应用软件，即可组成完整的单片机系统，因而更加方便、灵活，并且成本低，是目前在智能化仪器和控制中，应用最多的控制用微型计算机。

2. 可编程逻辑控制器（PLC）

可编程逻辑控制器（Programmable Logic Controller, PLC）是早期的继电器逻辑控制系统与微型计算机技术相结合的产物。如今的 PLC 几乎无一例外地采用微处理器作为主控制器，而采用大规模集成电路作为存储器及 I/O 接口，因而发展十分迅速。其可靠性、功能、价格、

体积都达到了比较成熟的境界，特别是卓越的技术指标和优异的抗干扰能力，在各个自动化领域，从柔性制造系统、机器人到工业局部网络都得到了广泛的应用。

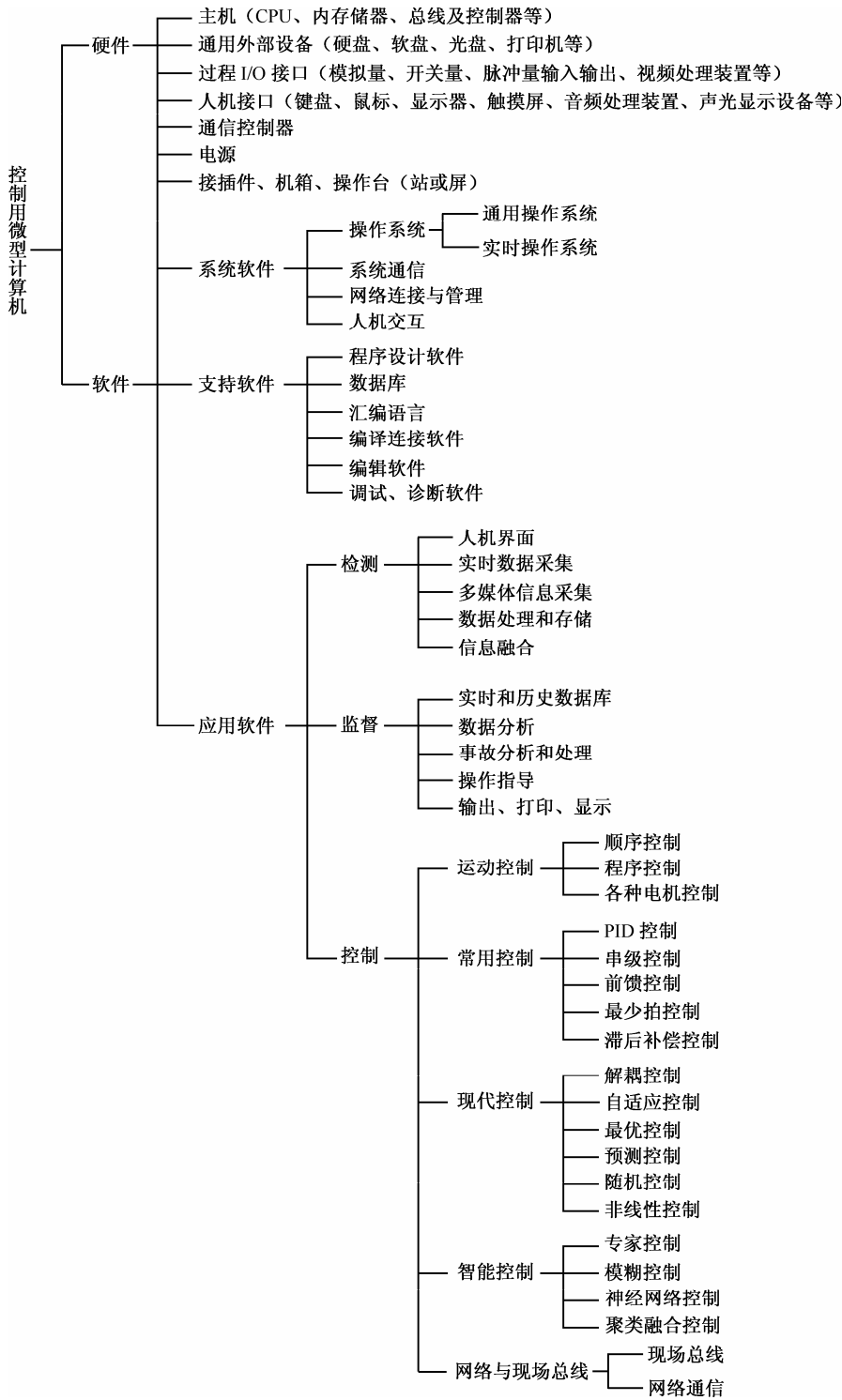


图 1.1 控制用微机的组成

从功能上, PLC 可以进行条件控制、定时控制、计数控制、顺序控制、A/D 及 D/A 转换、数据处理、通信与联网等功能。

在特点上, PLC 具有可靠性高、编程容易、组合灵活、输入/输出功能模块齐全、安装方便、运行速度快等优点。

近年来, 高档 PLC 微处理器大量采用多种模块和 DSP, 能力更是大大增强。

3. 工业控制计算机 (IPC)

工业控制计算机 (IPC) 一方面继承了个人计算机丰富的软件资源, 配以相应的工业应用软件, 使软件开发更加方便; 各种文字处理、报表打印、数据处理曲线、工业控制流程图、PID 柱形图等图形处理都变得简单。另一方面, 在结构上采用通用的底板总线插座系统, 将主板分成几块 PC 插件, 如 CPU 板、存储器板等, 把原 PC 电源改造成工业电源, 采用密封机箱, 并采用内部正压送风。

工业 PC 不但可以完成曾经流行于工业控制领域的 STD 总线工业控制机的检测、控制等各种功能, 而且使得工业 PC 的程序设计变得更加方便。随着 PC 的不断升级, 工业 PC 的性能也会相应提高。

1.1.2 控制用微型计算机的选用

随着应用规模的不同, 可参考下面的原则选择控制用微型计算机:

- 对于小型控制系统、智能化仪器及智能化接口, 可采用单片机模式, 若是新产品或用量较大, 也应尽量采用单片机开发;
- 中等规模的控制系统, 为了加快系统的开发速度, 应尽量选用现成的工业控制微机, 如 PLC、工业控制机等, 应用软件一般可自行开发;
- 大型的工业控制系统, 一般选用工业 PC 或专用集散控制系统, 软件可自行采用高级语言开发 (如 C 语言), 或使用现成的组合软件 (各种工业控制软件, 如组态软件等)。

1.2 微型计算机控制系统

微型计算机控制系统就是以微型计算机为控制器核心的自动控制系统, 开环和闭环控制系统的一般形式如图1.2、图1.3所示, 微型计算机控制系统的一般形式如图1.4所示。

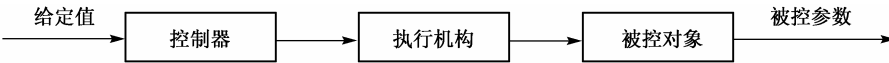


图 1.2 开环控制系统框图

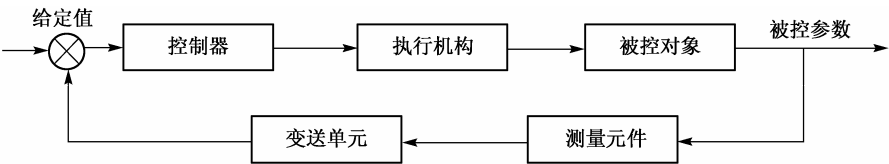


图 1.3 闭环控制系统框图

1.2.1 微机控制系统的组成

一般来说, 微机控制系统由微型计算机、接口电路、外部通用设备和被控对象等组成,

其典型结构如图1.5所示。被测参数的模拟量输入通道是：经传感器、变换器转换成统一的标准信号，再经多路开关分时送到 A/D 转换器，进行模拟/数字转换后的数字量通过接口送入计算机；在计算机内部，用软件对采集的数据进行处理和计算；输出的数字量通过 D/A 转换器转换成模拟量，然后经模拟量输出通道输出，再经反多路开关与相应的执行机构相连，以便对被控对象进行控制。

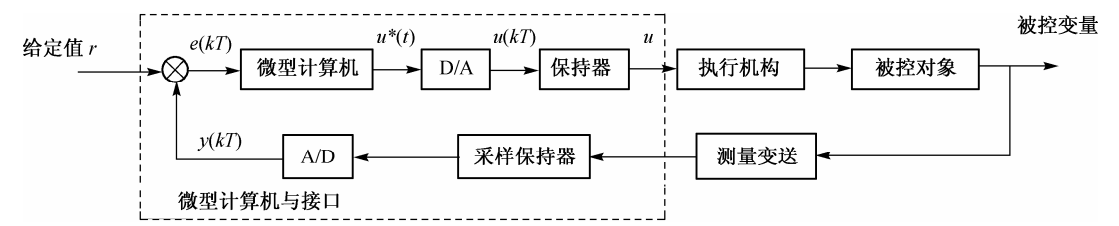


图 1.4 微型计算机控制系统的一般形式

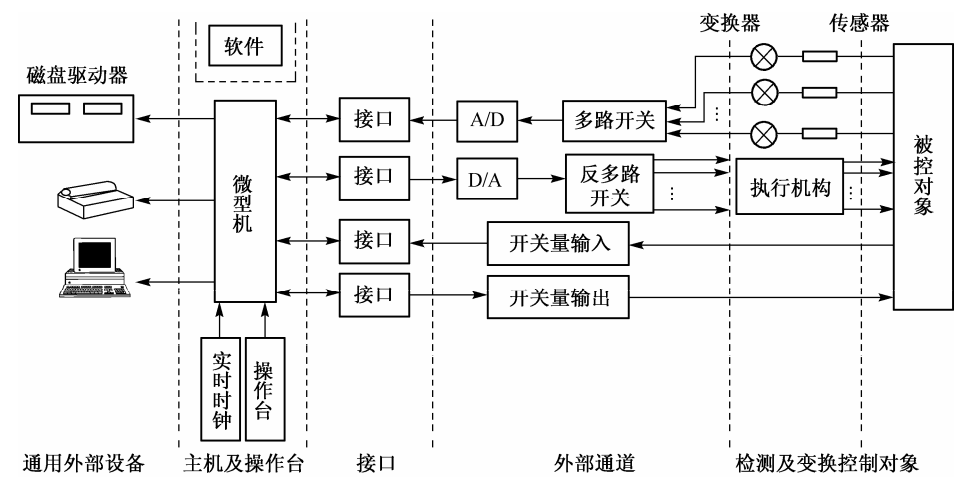


图 1.5 典型微机控制系统的组成

1.2.2 微机控制系统的工作过程与方式

1. 微机控制系统的工作过程

- ① 实时数据采集：对来自传感器或测量变送装置的被控量瞬时值进行检测和输入（包括对模拟输入量的 A/D 转换）。
- ② 实时控制决策：对采集到的被控量进行分析和处理，根据既定的控制策略或控制规律（例如，PID 控制、最优控制等），决定控制行为。
- ③ 输出实时控制信息：适时地对执行机构发出控制信号（包括对需要模拟输出量的 D/A 转换），实现控制和调节功能。
- ④ 信息管理：随着网络技术和控制策略的发展，信息共享和管理介入到较复杂的控制系统中，使整个系统达到更高的动态品质指标，及时监督被控参数并对异常状态做出迅速的处理。

2. 微机控制系统的工作方式

(1) 在线方式

被控对象和微机控制系统直接连接，并受其控制的方式称为在线方式（联机方式）。实时控制系统必然是在线系统，但只有符合实时性要求的在线系统才是实时系统。

(2) 离线方式

被控对象不和微机控制系统连接，也不受其直接控制，需要人进行联系并进行相应操作的方式称为离线方式（脱机方式）。

1.2.3 微机控制系统的功能与特点

1. 微机控制系统的功能

典型的微机控制系统应具有检测、监督和控制三方面功能。

(1) 检测功能

对被监测控制的对象参数进行采集与处理，并以一定的形式输出。

(2) 监督功能

将实时检测和人工输入的数据等信息进行分析、归纳、整理、计算等二次加工，做出诊断和预测，供有关人员参考、判断，其输出一般不直接作用于被控对象。

(3) 控制功能

在检测的基础上进行信息加工，根据预定的控制策略形成控制输出，直接作用于被控对象。

控制用微型计算机具有高速度、大容量和智能化的特点，可以实现上述三种功能的综合集成，构成完整的自动化监测控制系统；也可以针对实际应用的需要减少某些功能，以降低成本，降低复杂性，增强可维护性。

2. 微机控制系统的特点

(1) 实时性

实时性是系统对外界激励及时做出响应的能力，通常用“系统响应时间”来衡量。对随机发生的外部事件，微机控制系统应及时采集突发信号，并在所要求的时间内完成规定的任务。这是微机控制系统的关键特点，也是其重要的性能指标。

实时系统中，各种操作（中断、任务调度等）赋有不同优先级，如果高优先级的操作条件得到满足，系统将及时中断正常的运行，转而执行高优先级的操作。

不同的应用系统对实时性的要求不同。例如，温度系统由于温度变化的时间常数相对较大，系统响应时间一般为秒级；而电力监测控制系统由于要实时测量多处的电力负荷，分析故障，进行保护，如果响应时间过长，将会引起失控，甚至造成电力系统崩溃的严重后果，因而要求测控系统的响应时间是毫秒级甚至是微秒级。

与实时性有关的因素有如下两方面：

① 硬件方面。CPU 的时钟频率、字长、指令执行速度、中断优先级的处理电路等。

② 软件方面。操作系统对程序运行的实时调度和中断管理功能；程序的响应和处理速度等。合理地选用实时操作系统、模块的编程语言、程序结构、数据处理方法及控制算法，可以得到较好的效果。

(2) 可靠性

可靠性是系统无故障运行的能力。通常用“平均无故障运行时间”，即平均的故障间隔时间 MTBF（Mean Time Between Failures）来衡量。控制用微机的 MTBF 应在 10^5 小时以上。

保证系统可靠性的措施有如下四方面：

① 硬件方面。元器件经过严格的挑选、老化；印制电路板的设计、制造经过周密考虑；整机的安装、调试、测试等都有高水平的生产工艺和严格的质量标准。降低元器件的负荷率，特别是各 CPU 负荷率应小于 60%。重要的系统可采用双处理器或多处理器结构。增大系统内

存，减小内存覆盖面和虚拟内存，以提高数据的准确可靠性。各种与现场有关的信号都要经过隔离后再输入计算机。采用高可靠专用电源，适应较宽的电网电压波动，并可承受瞬间浪涌冲击。电源容量要有足够的富裕，例如，正常运行时只用到其额定值的 1/2；电源系统要有保证在电网不稳、电气干扰强烈的环境中可靠运行的防干扰措施。

② 软件方面。设计处理各种异常的检错纠错程序、适用的通信与容错支持程序、自诊断和自恢复程序。软硬件配合的容错设计，包括系统自诊断、自恢复、自动切换等。当系统内某些软硬件出现故障时，系统仍能完成预定的任务或某些重要的不允许间断的任务。采用标记检查、多级口令设置、加密等安全性控制方法来保证系统对自身文件和用户文件的存取合法性，防止对应用系统有意或无意的破坏。

③ 机械结构方面。通常采用全钢抗震防磁结构以适应电磁干扰强、温度过高或过低、湿度大、含腐蚀性物质、尘土多、颠簸振动大的恶劣环境；采用不锈钢材料或抗腐蚀涂层加工机箱，机箱内部保持正压，加滤网和双冷风扇抵御尘土和腐蚀性气体；加固机箱、采用避震器或安装可调整的弹性压条，以提高抗冲击和振动性能。

④ 系统组成方面。可选用下列冗余结构：

- 双机热备份：两套系统同时运行，正常情况下，备用机也在随时采集并保留实时数据，一旦主系统出现故障，切换装置立即把主控权切换到备用机使其成为主系统。双机热备份中还可采用维修快捷方便的带电拔插部件，以减小断电带来的影响，使系统出现故障后能尽早恢复正常状态。
- 双机冷备份：一用一备，工作组不正常时，备用组可立即投入运行。
- 部分硬件冗余：例如，对于寿命较短的磁盘（正常读写寿命平均为 5000 小时），可采用硬盘镜像方式，即双硬盘来保留同样的数据，也可采用磁盘阵列形成冗余结构，即使个别硬盘损坏，也不影响系统正常运转。

(3) 可维修性

可维修性包括进行维护工作的方便、快捷，并在最短时间内排除系统故障的能力。

保证系统可维护性的措施有以下两方面：

① 硬件方面。整体结构便于装卸和维修，便于及时更换；更换后，不必过多调整，就能保证系统运行状态和性能不受影响。有足够的信号显示出模板和部件的运行状态，并具有比较完善的诊断功能，便于准确查找故障。模块和部件可带电拔插。

② 软件方面。应用程序易阅读、易理解、易修改、易扩充、易移植。具有在线实时诊断程序，可以在不影响系统运行的情况下及时发现故障。

(4) 大量的现场交换信息

① 数据信息，包括模拟量、开关量和脉冲量的输入和输出。

② 多媒体信息，特别是图像信息已经进入控制领域，而且起着数据信息不能替代的作用。例如，视频信息经处理后可用以显示图像，也可以与数据信息融合后用做控制信号。

(5) 多种人机交互方式

除常规的键盘、鼠标、CRT 和 LCD 显示器外，还有触摸屏、专用键盘、大屏幕显示、语音等。

(6) 通信功能

多台监测控制计算机之间、相同类型或不同类型总线之间、计算机网络之间，实时、可靠地传递信息。

(7) 信息处理和控制算法

信息处理和控制算法的设计、开发、调试是研制微机控制系统难度最大的核心内容，也是开发设计人员的用武之地；软件的开发编制不但和采用的操作系统、软件开发工具有关，而且还和硬件（特别是接口部件）及生产工艺的要求有密切的关系。

近年来，针对各种检测控制对象和应用要求，以各种类型的控制用微机为核心，嵌入到应用系统中，构成了形形色色的自动化系统。千变万化的嵌入式应用对微机控制系统的性能提出了更高的要求。

1.2.4 微机控制系统的结构

随着微机控制系统功能和控制对象要求的不断提高，对于不同的被控对象及不同的控制要求，可采用不同的结构。

1. 操作指导控制系统

操作指导是指计算机的输出与各个控制单元不直接发生联系，而只对系统过程参数进行收集、加工处理，然后输出数据。控制动作实际上由操作人员接受计算机指示去完成。其原理如图1.6所示，属于开环控制型结构。

计算机根据一定的控制算法，依据输入通道（模拟量输入通道 AI 或开关量输入通道 DI）输入的由测量元件测得的信号数据，计算出供操作人员选择的最优操作条件及操作方案。操作人员根据计算机的输出信息，如 CRT 显示图形或数据、打印机输出等去改变调节器的给定值或直接操作执行机构。

操作指导控制系统是最早和最简单的计算机控制系统，其优点是结构简单，控制灵活而且安全。缺点是要由人工操作，速度受到限制，不能同时控制多个对象。它常用于进行数据检测处理及试验新的数学模型和调试新的控制程序等。

2. 直接数字控制（DDC）

直接数字控制（Direct Digital Control, DDC），是指计算机通过测量元件对一个或多个物理量进行巡回检测，经输入通道输入计算机，并根据规定的控制规律和给定值进行运算，然后发出控制信号直接去控制执行机构，使各个被控制量达到预定的要求。其系统原理图如图 1.7 所示。

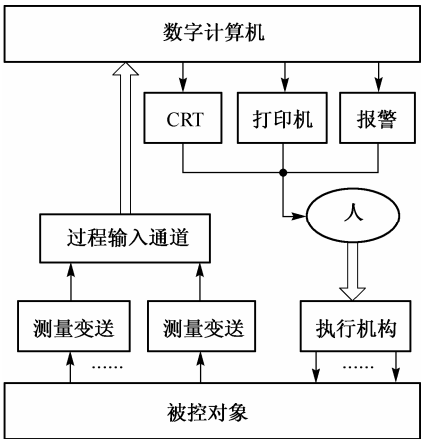


图 1.6 操作指导控制系统

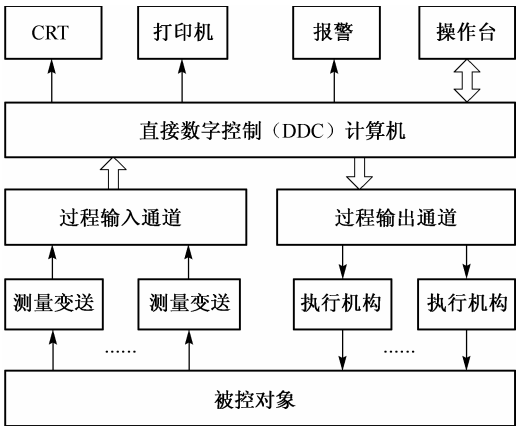


图 1.7 直接数字控制

在 DDC 系统中，计算机不仅能完全取代模拟调节器参加闭环控制过程，而且不需改变硬件，只通过改变程序就能有效地实现较复杂的控制规律，如前馈控制、非线性控制、自适应控制、最优控制等。由于计算机直接承担控制任务，且现场环境比较恶劣，干扰多，故要求 DDC 计算机可靠性高，实时性好，抗干扰能力强，能独立工作。所以一般选用微型机和工控机作为 DDC 的计算机。

DDC 系统是微型计算机作为数字控制器用于控制的典型系统，在各个领域都已获得广泛应用。

3. 监督控制（SCC）

计算机监督控制（Supervisory Computer Control, SCC）的构成如图1.8 所示。

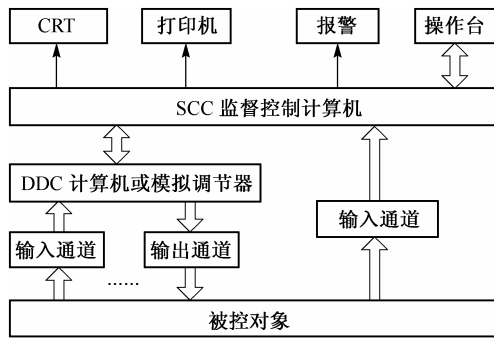


图 1.8 监督控制

在计算机监督控制系统中，由计算机按照描述生产过程的数学模型，计算出最佳给定值送给模拟调节器或者 DDC 计算机，最后由模拟调节器或 DDC 计算机控制生产过程，从而使生产过程处于最优工作状态。SCC 系统较 DDC 系统更接近生产变化的实际情况，不仅可以进行给定值控制，同时还可以进行顺序控制、最优控制及自适应控制等，是操作指导系统和 DDC 系统的综合与发展。

SCC 系统又可分为 SCC+模拟调节器控制系统和 SCC+DDC 控制系统。

在 SCC+模拟调节器控制系统中，由计算机系统对各物理量进行巡回检测，并按一定的数学模型，计算出最佳给定值并送给模拟调节器。此给定值在模拟调节器中与检测值进行比较后，其偏差值经模拟调节器计算后输出到执行机构，以达到调节生产过程的目的。当 SCC 计算机出现故障时，可由模拟调节器独立完成操作。

SCC+DDC 控制系统就是一个两级计算机控制系统，上级为监控级 SCC，下级为 DDC。监控级 SCC 的作用与 SCC+模拟调节器控制系统中的 SCC 一样，完成车间或工段等高一级的最优化分析和计算，并给出最佳给定值，送给下级直接控制生产过程。两级计算机之间通过接口进行信息交流，可由 SCC 级计算机完成 DDC 的控制功能。因此，大大提高了系统的可靠性。

在 SCC 中，由于 SCC 级计算机承担了先进控制、过程优化与部分管理任务，信息存储量大，计算任务重，故要求有较大的内存与外存和较为丰富的软件，所以一般选用高档微型计算机作为 SCC 级计算机。

4. 分布控制系统（DCS）

分布控制系统（Distributed Control System, DCS）也称为集散控制系统。在复杂的系统中，

分散的各设备基本独立地同时并行工作；随着微型机性价比的不断提高，分布控制系统（DCS）越来越多地代替了中小型计算机集中控制，避免了传输误差及系统的复杂化。在这种系统中，只有必要的信息，才传送到上一级计算机或中央控制室。分布式控制系统由现场控制级（DDC）、计算机监督控制级（SCC）和生产管理级（MIC）组成，其工作原理如图1.9所示。

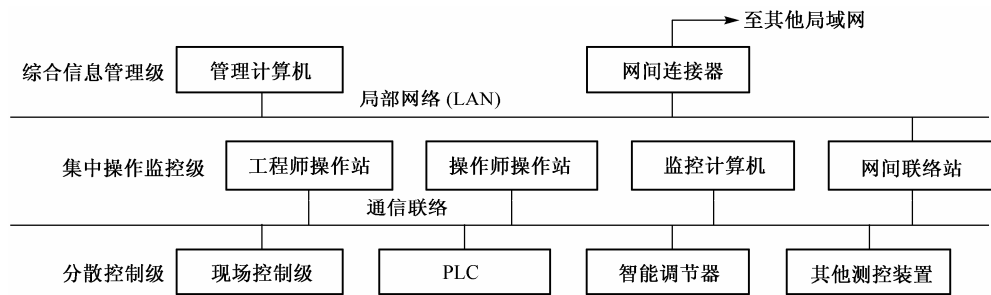


图 1.9 集散控制系统（DCS）

分散控制级（也称为现场控制级）是 DCS 的基础，用于直接控制系统。它由各工作站组成，每一工作站分别完成数据采集、顺序控制或某一被控制量的闭环控制等。现场控制级收集的数据供监控级调用，各工作站接收监控级发送的信息，并依此而工作。由于 DDC 系统的工作任务由各工作站分别来完成，局部的故障不会影响整个系统的工作，从而避免了集中控制系统“危险集中”的缺点。

监控级的任务是对系统进行监视与操作。监控级根据管理级的技术要求，确定现场控制级的最优给定量。监控级能全面地反映各工作站的情况，提供充分的信息，监控人员可以据此直接干预系统的运行。

管理级则是根据监控级提供的信息及任务的要求，编制全面反映整个系统工作情况的报表，审核控制方案，选择数学模型，制订最优控制策略，并下达命令。

DCS 综合应用了计算机技术、信号处理技术、测量控制技术、通信网络技术和人机接口技术，对系统进行集中监视、操作、管理和分散控制，具有通用性强，系统组态灵活，控制功能完善，数据处理方便，显示操作集中，人机界面友好，安装简单规范化，调试方便，以及运行安全可靠等特点。

5. 现场总线控制系统（FCS）

DCS 在 20 世纪 80 年代广泛应用于工业过程控制，但由于采用了“操作站控制现场仪表”的结构模式，系统成本较高，且各厂家生产的 DCS 有各自的标准，不能互联。

在 20 世纪 80 年代末、90 年代初出现了用于过程自动化、制造自动化、楼宇自动化等领域的将现场仪表和控制室仪表连接起来的全数字化、双向、多站的互联通信网络——现场总线技术，用数字信号取代模拟信号，提高了系统的可靠性、精确度和抗干扰能力，并延长了信息传输的距离。现场总线既是一个开放的通信网络，又是一种全分布的控制系统，是一种新型的网络集成自动化系统，它以现场总线为纽带，把挂接在总线上相关的网络节点组成自动化系统，实现基本控制、补偿计算、参数修改、报警、显示、综合自动化等多项功能。

现场总线控制系统（Field-bus Control System, FCS）将计算机网络通信与管理引入控制领域，被称为“21 世纪控制系统结构体系”。FCS 是一个开放式的互联网络，既可以与同层网络互联，也可以与不同层的网络互联，从而将系统中各种传感器、执行器及控制器通过现场

控制网络联系起来，通过网络信息传输完成传统系统中需要硬件连接才能传递的信号，完成各设备的协调，实现控制目标。

1.3 微机控制技术的发展与作用

1.3.1 控制理论与控制技术的发展

自动控制其实有着悠久的历史，在公元前 250 年古希腊人使用的滴水时钟中，就采用了水位调节器，其结构与至今仍在使用的浮子杠杆式水位控制器已经十分相近。近代控制理论与控制技术发展简史如表 1.1 所示。

表 1.1 近代控制理论与控制技术发展简史

项 目	年 代	概 要
近代控制技术起源	1778	Watt 蒸汽机离心调速器（比例控制）
控制理论起源	1868	Maxwell “On Governor”
电灯	1879	Edison 白炽灯
电力工业	1882	Edison 纽约珍珠街电站
PID 控制的构思	1922	Minorsky
PID 调节的原型	1936	Callender
PID 调节规则的提出	1942	Ziegler Nichols PID 控制器开始普及
采样定理	1949	Nyquist, Shannon
计算机诞生	1946	ENIAC 真空电子管式
Z 变换	1952	Hurwitz, Ragazzini, Zadan
计算机集中控制聚合装置	1959	Texaco Inc 第一套工业过程计算机控制系统
状态空间控制理论	1960	R. E. Kalman（滤波控制理论科学化）
计算机直接控制 DDC	1962	帝国化学工业公司
2 自由度控制系统概念	1963	I. M. Horowitz
模糊理论	1965	L. A. Zadan
模式预测控制	1960 年代后半期	
微型计算机诞生	1971	
微型计算机分散控制 DDC	1975	数字化控制开始，8~32 位字长的 DDC
知识工程	1977	
单环数字控制 DDC	1979	
H ∞ 控制	1980	
CRT 集中监视控制	1981	
人工神经网络	1986	
计算机测控（CIE）系统	1989	
遗传算法	1990	
Post 现代控制理论	1990	频域设计指标，状态空间设计方法
Light Sizing 系统	1995	PC 化、开放、标准化
国际标准现场总线	1998	
控制系统 IT 化	2000	

1.3.2 微机控制技术的发展

1946 年世界上第一台电子计算机诞生后不久, 20 世纪 50 年代中期便有人开始研究将计算机用于工业控制。1959 年, 世界上第一套工业过程计算机控制系统在美国得克萨斯州的一个炼油厂正式投运。该系统控制了 26 个流量、72 个温度、3 个压力、3 个成分。控制的主要目的是使反应器的压力最小, 确定反应器进料量的最优分配, 并根据催化作用, 控制热水流量和确定最优循环。

1971 年出现的微型计算机, 使微型计算机控制技术登上了历史舞台。微型计算机以其运算速度快, 体积小, 可靠性高和价格便宜的突出优点, 一方面使计算机控制系统从传统的、复杂的集中控制为主的结构形式转变为以微型计算机为核心, 综合了计算机技术、控制技术、通信技术与图形显示技术的分散控制系统, 不仅具有传统的控制功能和集中化的信息管理与操作显示功能, 而且还采用了高级语言进行实时控制, 有大规模的数据采集、处理功能, 以及较强的数据通信能力, 成为实现先进控制技术与生产管理的有力工具和手段, 从而使其在大型装置和系统的控制中占据了主导地位。另一方面微型计算机使得微机控制技术几乎不受价格和体积的限制, 渗透到各行各业的各种应用之中。即使是很小的控制任务, 也可以采用微型计算机来实现。

21 世纪将是微机控制技术飞速发展的新时期, 新的控制思想和新的控制技术将不断被过程机理、装置结构、生产环境和控制要求均十分复杂的装置和系统所采用, 微型计算机控制系统的微型化、操作系统平台的逐渐统一和网络技术飞速发展, 外延与内涵都将达到新的水平。在系统进一步完善, 功能更加丰富, 性能更加可靠的同时, 随着价格的不断下降, 应用也将更加普及。

1.3.3 微机控制技术的作用

在通往数字化、信息化、现代化的桥梁上, 微机控制技术是一块不可缺少的基石。

1. 微机控制技术是 IT 产业的一项重要支持技术

信息技术 (IT) 产业已经成为现代社会中的神经系统, 已经渗透到经济领域的各个层面——生产过程、营销过程、管理过程都离不开它。微型计算机控制技术则是 IT 产业的一项重要支持技术, 信息的采集、发送、传输、接收、处理、变换、显示都必须应用微机控制技术。

2. 微机控制技术是各行各业数字化、信息化的必备工具

在数字化、信息化带动各行各业现代化的过程中, 微型计算机控制技术起着不可缺少的关键作用。信息与控制的融合是 21 世纪的特征, 数字化、信息化是各行各业现代化的标志, 而微机控制技术正是实现数字化、信息化的必备工具。从数字化部队的各种随身装备到航空航天装置, 从各种迷你型数字化视听、数码世界等到交通、化工、电力、冶金、纺织等各行各业和各项大型枢纽工程, 都可以看到微机控制技术突飞猛进的普及和应用。

1.3.4 微机控制技术的前景

微型计算机控制技术是计算机技术、自动控制技术、人工智能技术、微电子技术、检测与显示技术、通信与网络技术高速发展的结晶, 在 21 世纪将达到更高的水平, 得到更普遍的应用。

1. 成熟先进技术的推广应用

微型计算机控制技术中，很多成熟技术将得到发展和推广。

(1) 普及应用可编程序控制器（PLC）

近年来，具有 A/D、D/A 转换器和 PID 调节等功能的 PLC 的出现，使得 PLC 的功能有了很大提高，中、高档 PLC 可以将顺序控制和过程控制结合起来，实现对生产过程的控制，并具有很高的可靠性，从而将得到广泛的普及和应用。

(2) 广泛使用智能调节器

智能化调节器不仅可以接受 4~20 mA 标准电流信号，还具有各种异步串行通信接口，可与上位机连成主从式测控系统。

(3) 采用新型的 DCS 和 FCS

发展以位总线（Bitbus）、现场总线（Fieldbus）等先进网络通信技术为基础的 DCS 和 FCS 控制结构，并采用先进的控制策略，向低成本综合自动化系统的方向发展，实现计算机集成制造系统（CIMS）。

2. 智能控制技术的深入研究和发

经典的反馈控制、现代控制和大系统理论在应用中遇到不少难题。因为这些控制系统的设计和分析都是建立在精确的系统数学模型的基础上的，而实际上难以获得精确的数学模型；而为了提高控制性能，整个控制系统变得极其复杂，增加了设备的投资，降低了系统的可靠性。智能控制是用计算机模拟人类智能的一个重要领域。其中，学习控制、模糊控制、人工神经网络控制等智能控制方法为微机控制系统提供了更加广阔的应用前景。随着多媒体计算机和人工智能计算机的发展，应用智能控制技术的微机控制系统，将大大推动科学技术的进步和提高自动化系统的水平。

习题与思考题

- 1 回顾三次工业（科技）革命的历史，找出它们的标志性发明，并由此进一步体会自动控制，特别是微机控制技术的重要作用。
- 2 控制用微型计算机有哪些特点？它由哪几部分组成？
- 3 简述微机控制系统的功能与特点，微机控制系统的不同工作方式和结构分别适用于怎样的控制对象和要求？
- 4 在数字化、信息化、现代化的过程中，微机控制技术有哪些重要作用？
- 5 你的身边有哪些设施应用了微机控制技术？试举一例，并分析其控制方案。
- 6 你认为在哪个领域中微机控制技术的应用将得到较大发展？简述该领域的现状和技术发展趋势。

第2章 常用数据结构和数据处理

信息与物质、能量并列为宇宙三大要素，信息的表示是计算机科学的基础。对大多数计算机程序来说，信息的存储和检索可能比完成运算更为困难和重要。程序必须从存储空间和运行时间这两个角度来组织信息，才能进行高效可靠的信息处理。研究数据结构和算法以有效地支持程序的实现，是计算机科学的核心问题之一。

微机控制系统处理信息的范围日趋扩大和复杂，不但要进行数值计算和数据的输入输出，还需要处理大量的非数值数据。为了设计高效率、高可靠性的程序，不但要掌握编程技术，还要研究程序所加工的对象——数据及其处理方法，只有彻底理解这些隐藏在高效程序设计后面的一般原理，才能自如地组织和运用数据。

2.1 常用数据结构

2.1.1 数据结构的基本概念

数据 (Data)：所有能输入到计算机中并被计算机程序处理的符号的集合。

微机控制技术中，不仅要处理数值、字符，还要处理更为复杂的非数值数据和实时数据，例如，符号、表格、图表等，以及图像等多媒体信息。

数据元素 (Data Element)：数据的基本单位。有时一个数据元素可由若干个数据项 (Data Item) 组成，数据项是数据的最小单位。

例如，学生成绩登记表中，每个学生的情况，即表中的每一行就是一个数据元素；而每一格如学号、姓名、各次考试的成绩就是一个数据项。

数据对象 (Data Object)：具有相同特性的数据元素的集合，是数据的一个子集。

例如，整数的数据对象是集合 $N = \{0, \pm 1, \pm 2, \dots\}$ ，字母字符的数据对象为集合 $C = \{A, B, \dots, Z\}$ 。

被计算机加工的数据元素之间存在着某种联系，这种相互之间的关系，通常称为结构。

数据结构 (Data Structure)：包括数据的逻辑结构和物理结构。数据的逻辑结构是数据元素之间的逻辑关系；数据的物理结构（或称为存储结构）是数据元素在计算机存储器内的表示及配置。

数据结构不但研究数据本身的特性，而且研究数据之间存在的关系和数据的组织，是一个二元组

$$\text{Data-Structure} = (D, R) \quad (2.1)$$

其中， D 是数据元素的集合， R 是 D 中数据元素之间所存在的关系的集合。

如果数据集合 D 中的数据元素之间存在着不同的关系集合 R_1 和 R_2 ，那么数据结构 $DS_1 = (D, R_1)$ 和 $DS_2 = (D, R_2)$ 是两个不同的数据结构。

数据结构只研究数据之间的关系，并不涉及元素的具体内容。例如，矩阵

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} = [a_{ij}] \quad (2.2)$$

就可以用行号 i ($i=1,2,\cdots,m$) 和列号 j ($j=1,2,\cdots,n$) 来表示集合的关系 R 。

每一个数据结构都有相关的代价和效率的权衡，为了确定对于一个新问题最合适的数据结构，需要评估其有效性。常用数据结构如下。

1. 顺序结构

顺序存储结构中，数据从某个存储地址开始依次存放在连续的存储单元中。

(1) 线性表

线性表是 n 个数据元素的有限序列，除了表头和表尾外，表中每个数据元素都有且只有一个直接前趋，有且只有一个直接后继。可表示为

$$(a_1, a_2, \cdots, a_i, \cdots, a_n) \quad (n \geq 0)$$

其中， a_i ($i=1,2,\cdots,n$) 是数据元素，其下标 i 表示元素的序号，代表元素在线性表中的位置。 n 是表中元素的个数，定义为表的长度。若 $n=0$ ，则为空表。

在线性表中，每个数据元素的位置取决于其序号，假设每个元素占用 L 个存储单元，则第 i 个元素 a_i 的存储地址 $\text{LOC}(a_i)$ 表示为

$$\text{LOC}(a_i) = \text{LOC}(a_1) + (i-1) \times L \quad (2.3)$$

其中， $\text{LOC}(a_1)$ 是线性表中第一个元素的存储地址，也称为表头地址。

运算和运算规则是构成一个数据结构不可缺少的部分，对线性表可以进行下列运算：

- 确定线性表的长度 n ；
- 存取线性表的第 i 个数据元素，检查或修改数据值；
- 删除第 i 个元素；
- 在第 i 个和第 $(i+1)$ 个数据元素之间，插入一个新的数据元素；
- 将一个线性表拆成两个或两个以上的线性表；
- 将两个或两个以上的线性表合并成一个线性表；
- 按某个特定值查找数据元素；
- 对线性表中的数据元素按某个数据值递增（或递减）的顺序重新排序；
- 复制一个线性表。

一般情况下，只需进行其中一部分运算。线性表最常用的运算是查找、排序、删除和插入。

(2) 数组

数组 (Array) 是线性表的简单推广，其中每个元素由一个数值和一组下标组成。例如，一个 $m \times n$ 的矩阵式可以看成是一个二维数组，其中每个元素 a_{ij} 是一个值，同时又和一个二维空间的坐标 (i, j) ($i=1,2,\cdots,m; j=1,2,\cdots,n$) 相对应。

反之，线性表是数组的特例。例如，数组中的一行或一列就是一个线性表。

数组通常有以下运算：

- 给定一组下标，查找与其对应的数据元素；
- 给定一组下标，存取或修改与其对应的数据元素。

(3) 堆栈

堆栈 (Stack) 是特殊结构的线性表, 限定数据元素的插入或删除 (包括存取) 只能在表的一端进行。允许插入或删除的一端称为栈顶 (Top), 其相反的另一端则称为栈底 (Bottom)。表中元素以 a_1, a_2, \dots, a_n 的顺序进栈, 以相反的顺序 a_n, \dots, a_2, a_1 出栈, 堆栈正是因后进先出 (LIFO) 的特点如实际生活中的货物堆栈而得名的。

每个堆栈都有一个栈指针 sp 指示栈顶位置, 当 $sp = 0$ 时, 称为空栈; 当 $sp =$ 堆栈的长度 n 时, 称为满栈。每次进栈, 在栈顶加入一个新的数据元素, sp 加 1; 反之, 每次退栈, 删去栈顶一个数据元素, sp 减 1。满栈 ($sp = n$) 时再进栈或空栈 ($sp = 0$) 时再退栈, 会发生上溢或下溢错误。

除了地址增加型 (每存放一个数据, 存储地址增加) 堆栈外, 还有地址减少型 (每存放一个数据, 存储地址减少) 堆栈, 如图 2.1 所示。

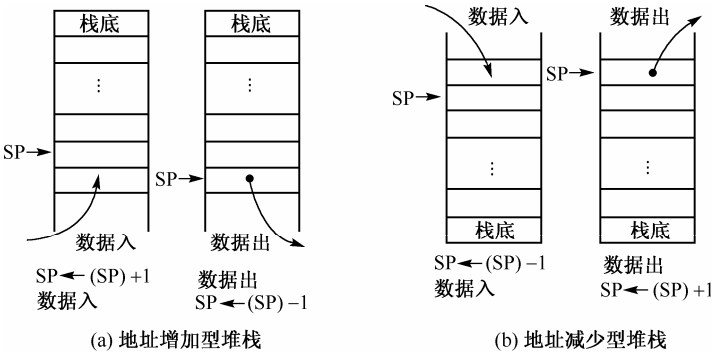


图 2.1 两种类型堆栈

堆栈的运算有进栈、退栈和读栈顶一个数据元素。

除了暂存数据之外, 经常采用称为堆栈的后进先出 RAM 缓冲区来保护现场, 处理过程的嵌套、调用, 例如, 子程序的调用及返回处理, 中断断点的保存和恢复等。在微机控制的实际应用中, 堆栈既可以应用在正常有序的程序运行中, 也可以用于处理非正常的程序运行中, 例如突发事件和故障, 还可以用来在程序之间交换数据。

(4) 队列

队列 (Queue) 也是一种特殊的线性表, 和日常生活中的排队一样, 表中元素以 a_1, a_2, \dots, a_n 的顺序进入, 以同样的顺序出去。因为从队头取出的是队列中最先进入的元素, 所以是先进先出表 (FIFO), 如图 2.2 所示。通常有两种运算: 加入和删除。

队列限定所有数据元素的加入在表的一端进行, 而读出在表的另一端进行。在先进先出表 (FIFO) 中, 允许加入的一端称为队尾 (Rear), 允许读出的一端称为队头 (Front)。每个队列需设立两个指针, 队尾指针 RP 指向队列的最后一个元素之后的一个空位, 表示下一个要加入队列的元素的位置; 队头指针 FP 指向队列最前面的一个元素, 表示要取出的元素的位置。

当加入一个元素时, 把它存入 RP 所指出的位置, 然后把 RP 加 1; 当取出一个元素时, 按 FP 所指取出队列的第一个元素, 然后把 FP 加 1。当 $(RP) = (FP)$ 时, 队列为空, 即没有元素。初始时, $(RP) = (FP) =$ 队列的初始地址。

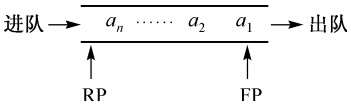


图 2.2 队列示意图

计算机中的队列，如果也像日常生活中一样，当取出一个元素时，就把队列中其他元素向队头方向移一步，那么将耗费大量的操作时间；但不这样做，又会使队列在存储器中的位置逐渐后移，以至于无处再插入新的元素，而前面的空位却越来越多，造成假溢出。

循环队列是一个有效的解决方法。它可以想象成是一个圆环形的存储器阵列：当 RP 增加到超过循环队列的长度 N 时，置 RP 为初始值 0 ，重新从初始位置开始存入元素；当 FP 增加到超过循环队列的长度 N 时，也置 FP 为初始值 0 ，如图 2.3 所示。

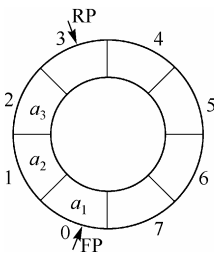


图 2.3 循环队列

判断循环队列是空还是溢出的方法有两种：

- 设立一个标志位，以区别队列状态；
- 若 RP 指针变化后等于 FP ，则队列为满；若 FP 指针变化后等于 RP ，则队列为空。

循环队列的加入流程和读出流程分别如图 2.4 和图 2.5 所示。

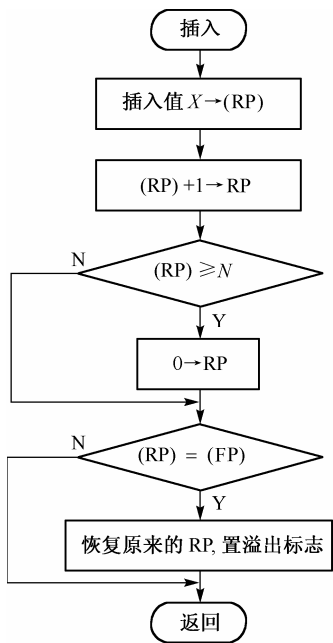


图 2.4 循环队列的加入流程

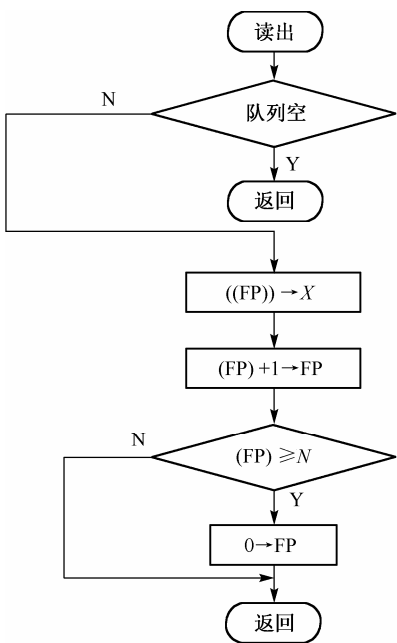


图 2.5 循环队列的读出流程

循环队列可用于输入输出环形缓冲器、滑动平均值计算和其他需使用队列的场合。

在前面介绍的顺序结构中，线性表、数组为静态顺序结构，堆栈、队列为动态顺序结构。由于它们都要求连续的存储单元来顺序地存放数据元素，因而共同的缺点是，当插入或删除时，要移动大量的数据元素，并浪费操作时间；而若留有扩展余地，又将浪费存储空间。为了克服这些缺点，可以采用链形结构，简称链表。

2. 链表

链表由于其组成像链条而得名，它由若干个结点组成，每个结点有两个域：存放数据元素的数据域和存放下一个结点数据域首地址的指针域。通过指针域把各结点按要求的顺序连接起来组成一个环环相扣的表，在逻辑上是有序的；而各结点在存储器中的物理位置可以任意配置，可能是无序的。

链表中，设置头指针（Head），以确定第一个结点的数据域首址；最后一个结点的标志通常是将其指针域设置为“空”。单链表如图 2.6(a)所示。

链表的运算通常有三种：查找、插入和删除结点。

(1) 链表的查找

给出链表中某结点的存储地址，便可从头指针开始顺序查找到该结点。

如果给出的是数据项的关键字 x ，则需要从第一个结点开始，逐个比较结点的数据项的关键字是否等于给定值 x ，相等说明查找成功，否则继续向后找下一个结点，直至查到或确认查不到（指针值等于 0）为止。

采用循环链表或双重链表可以加快查找。

把单链中最后一个结点的空指针域改成指向第一个结点的首址，就构成循环链表。其优点是可以从任何一个结点开始查找所需结点，但仍只能单向查找，如图 2.6(b)所示。

双重循环链表可以双向查找。双重循环链表中每个结点有三个域：左指针域、数据域和右指针域。左指针域用来链接其前趋结点，右指针域用来链接其后继结点，如图 2.6(c)所示。

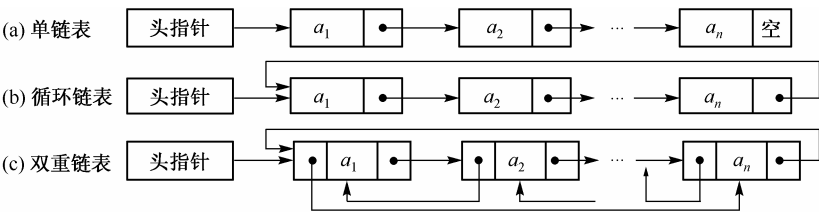


图 2.6 三种链表结构示例

(2) 链表的插入与删除

链表的插入或删除，只需改变结点的指针域，而不必变更其物理位置，因而进行文件的组织、增删非常方便，常用于存放大量增删频繁的数据和信息。例如，微机控制系统中众多的控制模块、运算模块、顺序逻辑模块所对应的数据区，以及大量的操作信息、历史记录信息等。

实际上，为了对一个由起始指针 **FIRST** 和若干个结点组成的线性链表进行插入或删除，还需要留出一定的存储空间，并将这些空间组成一个由指针 **FREE** 指向第一个结点的链表，称为自由表，如图 2.7 所示。当链表需要插入一个结点时，取出自由表中 **FREE** 指针指向的结点插入到链表中去；在链表需删除一个结点时，把该结点加入到自由表的第一个结点前，成为 **FREE** 指向的第一个结点。一般来说，几个链表可合用一个自由表，这样可提高存储空间的利用率。

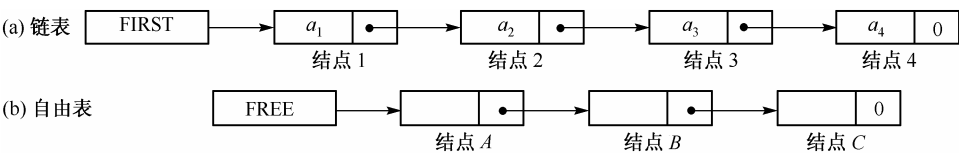


图 2.7 链表的实际组织

- ① 链表的插入。在链表结点 K 和结点 $K+1$ 之间插入新结点的操作步骤如图 2.8 所示：
- 先判断 **FREE** 指针是否为 0，若为 0 则表示自由表空，不能用于插入，否则进行以后的操作；

- 保存结点 K 的指针值（即结点 $K+1$ 的地址）；
- 把 **FREE** 指针值（即结点 A 的地址）写入结点 K 的指针；
- 读出新结点的指针值（即结点 B 的地址），并写入 **FREE** 中；
- 把保存的原结点 K 的指针值写入新结点的指针；
- 数据存入新结点。

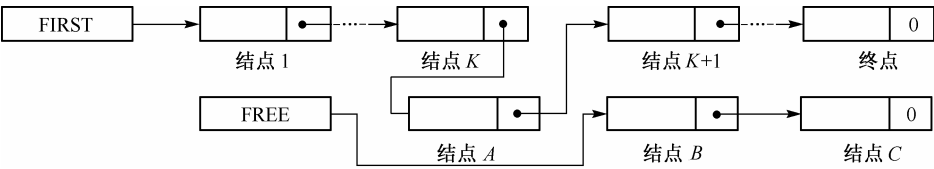


图 2.8 结点 A 插入链表

② 链表的删除。将链表的结点 2 删除后放入自由表中的操作步骤如图 2.9 所示；

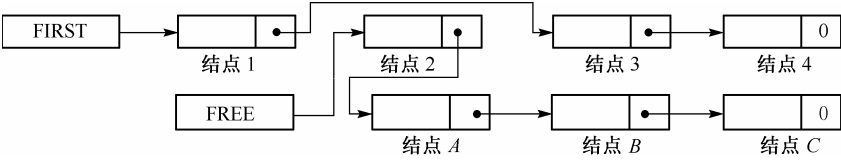


图 2.9 从链表中删除结点 2

3. 树

计算机处理的许多问题，如果采用非线性结构往往比一般线性结构更为明确、方便。例如，用树结构来管理具有层次关系或上下级关系的数据、信息和文件。

(1) 树的结构

非线性结构中，至少存在一个结点（或数据元素）有不只一个前趋或后继。树就是一类重要的非线性结构。

树形结构，或简称树（Tree），因其形状很像树的倒置而得名（如图 2.10 所示）。

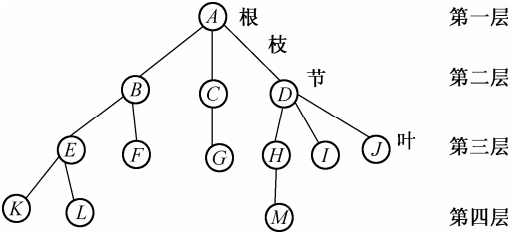


图 2.10 树的示例

为了形象地描述树中各结点之间的层次关系，把最高结点 A 称为树“根”，结点之间的连线称为树“枝”，具有下枝的结点称为树“节”，不具有下枝的结点称为树“叶”。结点的下枝数称为“度”，树的“度”是树中各结点的最大度。度 = 2 的树也称为二叉树。

也常用家族术语来形象地描述树中各结点之间的层次关系，例如，在图 2.10 中，结点 A 是结点 B, C, D 的父亲，结点 B 是结点 E, F 的父亲，结点 C 是结点 G 的父亲，结点 A 又是结点 E, F, G 的祖父，还可以使用儿子、孙子等术语。结点的层次从根算起，树中结点的最大层次称为树的深度。

(2) 树的遍历

以一定顺序对树的每一结点做一次访问的过程称为树的遍历。常用的遍历方法有：

- 前序遍历：首先访问根结点，然后按照上层结点、左子树、右子树的次序访问各棵子树，以图 2.10 的树为例就是——*ABEKLFCGDHMIJ*；
- 后序遍历：按照左子树、右子树、上层结点的次序，先访问各棵子树，然后访问根结点——*KLEFBGCMHIJDA*；
- 层次遍历：首先访问根结点，然后依次逐层访问以下各层上的结点——*ABCDEFGH IJ KLM*；
- 获得树中所有叶结点：如果树中只有一个结点，那么此结点就是此树的叶子结点；否则，树中的叶子结点就是根结点的各棵子树的叶子结点——*KLFGMIJ*。

(3) 二叉树的表示和转化

用双指针链表来表示二叉树，可以很方便地进行插入、删除和查找，并且比用有序表节省很多存储单元。树的深度越深，就越节省。图 2.11 是用链表表示二叉树的例子。

用一种完整而有规律的走法遍历二叉树，可以得到二叉树结点（信息）的线性排列。例如，可以用图 2.12 的二叉树来表达一个算术表达式

$$e/f-(c\times d-(a+b)) \tag{2.4}$$

其中，根和节表示运算符，而叶表示运算数。从叶开始逐层向上寻找对应的父亲即运算符。每上一层就进行一次运算，直至根结点。

后序遍历该树，可得到字符串 *ef/cd×ab+--*。这个线性排列（称为表达式的后缀表示或称为逆波兰记号）清晰地表达了树的含义。

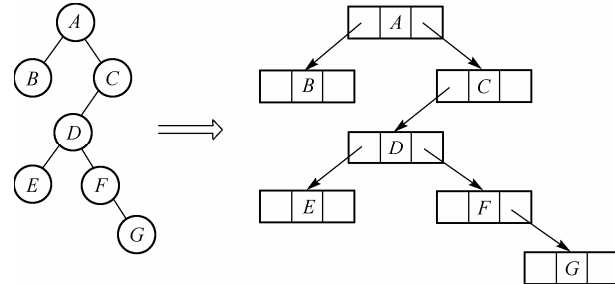


图 2.11 二叉树的链表表示

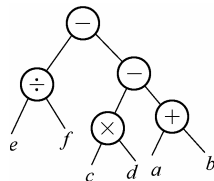


图 2.12 表达式 $e/f-(c\times d-(a+b))$ 的二叉树

一般树结点的度各不相同，用链表表示时，若用不定长的结点，则运算十分复杂；若用定长结点，又将造成大量指针域的浪费。为了提高效率，可以把一般树转化为二叉树。

转化步骤如下：

- 在兄弟结点之间加一条连线；
- 对每个结点，除掉其最左的子结点之外，抹掉该结点与其余子结点之间的连线，并且将每一层结点与最左边的子结点连线变成垂直，兄弟结点之间的连线放成水平；
- 以根结点为轴心，将整棵树顺时针转 45° 。

图 2.13 就是按照以上步骤得到的与图 2.10 的树相对应的二叉树。在这棵二叉树中，每一结点的左子结点是原树中的儿子，右子结点则是原树中的兄弟。

树是非线性结构，所以，用树表达一些问题往往比一般线性结构更清晰。二叉树具有良好的性质，在排序、表达式编译、判断及二进制编码及信息检索等方面都有很大用处。这些不同应用的树，都是按一定规定构成的，往往赋以各种名称，由此形成了构造特定树、遍历二叉树及计算树的路径长度（从一个结点到另一个结点之间的树枝数称为此对结点之间的路径长度）等算法。

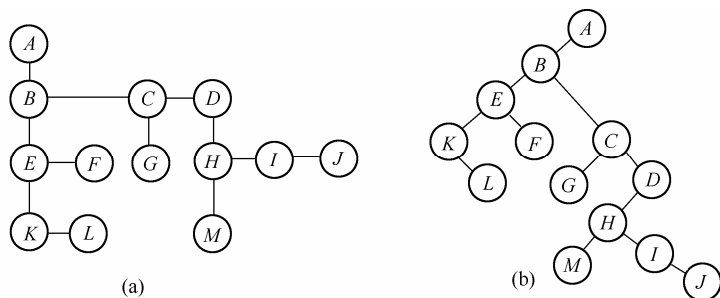


图 2.13 与图 2.10 树相对应的二叉树

2.1.2 控制系统的特殊数据结构

微机控制系统中，往往有多个进程并发执行。为了协调、调度这些进程，需要采用一些特殊的数据结构，以实现进程之间的通信并使它们同步。

1. 信号量及其操作

(1) 信号量

信号量是一个与队列有关的整型变量，经常在操作系统中表示资源的物理实体。例如，打印机是否被调用，某个数据公用区是否正在被修改，等等。

信号量的值仅能由 P 和 V 操作来改变。进程 q 对信号量 s 的 P 操作和 V 操作，写成 $P(s)$ 和 $V(s)$ 。

$P(s)$: s 值减 1。

此时，若 $s \geq 0$ ，则进程 q 继续执行；

若 $s < 0$ ，则 q 进程被封锁，并且被放入一个与 s 相联系的等待队列 Q 中。

$V(s)$: s 值加 1。

此时，若 $s > 0$ ，则 q 进程继续执行；

若 $s \leq 0$ ，则从与 s 相联系的等待队列 Q 中移出一个过程，把它解除封锁，即唤醒该进程，并把它插入就绪队列中， q 进程继续执行。

要注意的是 P 和 V 的操作在执行期间不能被割断。

$s > 0$ 表示某类可用资源的数量， $P(s)$ 操作意味着请求分配一个单位的资源，所以令 $s = s - 1$ ；若 $s < 0$ ，就表示已无资源，故该进程应被封锁，并进入等待队列；此时 s 的绝对值等于队列 Q 中等待分配资源的进程数目。

$V(s)$ 操作意味着释放一个单位的资源，所以令 $s = s + 1$ ，此时，若 $s < 0$ 就表示队列 Q 中仍有因请求该资源而被封锁的进程，应该把队列 Q 中的第一个进程唤醒，使之移到就绪队列。

信号量分为两种：

公用信号量，用来管理一组进程，实现进程的互斥。被管理的进程均可对它进行 P 和 V 操作；

私用信号量，其初始值应为 0 或某个正整数。一组并行进程中，只允许各进程对各自的私用信号量进行 P 操作或 V 操作。

(2) 用信号量实现进程的互斥

进程在并发执行中需要合理分配资源。很多资源一次只能被一个进程利用，称为临界资源，例如，某一设备，如输入机、打印机、磁带机等，或者是某一数据库如一些变量、数据、表格、队列等。进程访问临界资源的程序段称为临界区，诸进程进入临界区时互斥。互斥包括下列内容：

当无进程处于临界区时，允许一进程立即进入临界区；

当有一进程处于临界区内时，其他试图进入临界区的进程必须等待；

当有一进程离开临界区时，若有等待进程，则允许其中一个进入临界区。

用信号量的 $P-V$ 操作就可以实现上述进程的互斥。

(3) 用信号量实现进程的同步

在多个过程并发执行的控制系统中，除了临界区互斥问题外，更多的是异步进程环境下的同步问题，即进程必须交换信息，使各进程的运行协调。

【例 2.1】用信号量实现计算与打印进程之间的同步。

设 A 进程进行计算并把结果送入一缓冲区， B 进程将该缓冲区中的计算结果打印出来。这两个进程各自独立，但又必须交换信息以互相制约： A 应先算出结果并送到缓冲区，否则 B 就不能打印； B 接到计算结果并打印后， A 才能再次运行，否则会冲掉上一次的计算结果。这种制约就称为进程之间的同步。同步和互斥一样，都属于进程之间的通信。

用 A 进程的私用信号量 SA 表示缓冲区中是否已有可供打印的计算结果， B 进程的私用信号量 SB 表示缓冲区中的计算结果是否已被打印进程 B 取走。用 SA 和 SB 的 P 和 V 操作来实现 A 进程和 B 进程同步的描述如下：

A 进程

L_1 : 计算下一结果并送到缓冲区

$V(SA)$: $SA+1$, 缓冲区有值

$P(SB)$: $SB-1$, 请求打印

若 $SB < 0$, 打印机被占等待

B 进程

L_2 : 把缓冲区中结果取走并打印

$P(SA)$: $SA-1$, 从缓冲区取走

若 $SA < 0$, 无数据等待

$V(SB)$: $SB+1$, 释放打印机

示例中，只允许各进程对自己的私用信号量进行 V 操作。

2. 信息缓冲区和信箱

进程之间的同步通信可包括两种类型：信号同步和信件同步。

信号同步比较简单，发送者只须给对方一个简单的信号，而接收者的响应也只须给一个简单的回答信号。信号的含义，双方已经约定。

信件同步要复杂些。进程之间交换的是一组信息，因而需要有高级的通信手段。信息缓冲区和信箱就是微机控制系统中常用的进程通信的数据结构。

(1) 信息缓冲区

信息缓冲区可以是链表结构，进程通信的一个基本单位包括下列信息域：

- 指向发送进程的指针 $Sptr$;
- 指向下一个信息缓冲区的指针 $Nptr$;
- 消息长度 $Size$;
- 消息正文 $Text$ 。

在进程不多、通信不频繁的微机控制系统中，也可以用循环队列形成信息缓冲器，发送进程将信息存入缓冲器队尾，而接收进程从队头移出信息。循环队列缓冲器的存储分配是固定的，没有链表结构那样灵活，但在系统不安排自由存储空间时处理方法比较简单。

(2) 信箱

在信件通信中，还可以采用信箱通信方式。信箱这种数据结构在逻辑上分成两部分。一部分是信箱头，用以描述信箱体。另一部分是信箱体。信箱由若干个格子组成。每个格子存放一封信件。格子的大小根据信件的高低来定，格子数也可根据需要决定，最简单的信箱是

单方向传送的。只要有空格就可由发送进程向里投放信件。只要格子中有信件，接收信件就可以从中取走一信件。信箱中信件的管理要注意溢出的问题。当格子放满信件时，再投入信件，就称为“上溢”；当信箱空时，再去索取信件，就称为“下溢”。溢出时就必须阻塞相应进程，以使发送进程和接收信件的进程同步。

在实时系统中，两进程之间的通信往往需要有回答，信箱也可以演变为双向的方式：发送进程每发出一封信，就等待一封回答；而接收进程每收到一封信后，就发回一封回答。

2.2 采样与数据处理

微机控制系统中，对信号的测量过程一般是：先用传感器把非电信号转换成电信号，然后用 A/D 转换器按一定周期对模拟信号采样、量化，变成数字信号。对读入计算机的数据，需要进行各种处理。与科学计算不同，微机控制系统对数据处理方法的实时性要求较高，如计算时间应尽可能短，占用资源应尽可能少；而对精度要求相对来说较低，数据范围一般也是已知的。

2.2.1 采样周期的选择

采样周期 T 或采样频率 ω 是微机控制系统的重要参数，在系统设计时必须考虑各种因素，选择合适的采样周期。若采样周期取得大，则可以允许系统采用比较复杂的算法，或者降低对计算机运行速度、A/D 及 D/A 转换速度的要求，以降低系统成本；但采样间隔内控制作用不变。过大的采样周期会降低系统的性能。

1. 采样周期对系统性能的影响

(1) 采样周期对系统稳定性的影响

微机控制系统里，采样周期对闭环系统的稳定性和性能有很大的影响。当系统参数一定时，可以确定使系统稳定的最大采样周期 T_{\max} ，实际选用的采样周期则应比 T_{\max} 小得多。

(2) 采样周期对信息丢失的影响

对信号的采样将会丢失采样间隔之间的信息，为了保证系统的输出能准确复现系统的输入信号，得出正确的误差以进行控制，根据采样定理，假定输入及反馈信号的最大频率为 ω_{\max} ，则采样频率 ω_s 应满足 $\omega_s \geq 2\omega_{\max}$ 。

实际应用中，信号的最大频率 ω_{\max} 难于确定，而有些信号（如常用阶跃信号）所含的频率很高又常常发生变化，为了减少频率混叠现象，常用的方法是：

① 根据被控对象全部特征根中的最高频率 ω_{\max} 选取。系统中被采样的信号经常是被控对象的输出信号，其特性由被控对象的特征根决定，因此可以认为系统输出中所含的最高频率分量是由被控对象特征根中的最高频率来决定的。一般要求 $\omega_s \geq (4 \sim 10)\omega_{\max}$ 。

② 根据系统闭环频带 ω_b 选取。闭环控制系统的频带是有限的。频率高于系统闭环频带的被控对象输出分量在整个输出信号中所占的比例很小，因此可以认为闭环频带是信号的最高频率。一般要求 $\omega_s \geq (4 \sim 10)\omega_b$ 。

在微机控制系统中，使用前置滤波器，是放宽对采样周期的限制，即允许选用相对较低采样频率的有效措施。

(3) 采样周期对系统抑制干扰能力的影响

为了及时抑制干扰对输出的影响，应该按照干扰信号中高于最高频率 ω_{\max} 2 倍以上的原则

来选择采样频率 ω_s ，但通常干扰的频率较高，实现这一点较为困难。微机控制系统中，通常将干扰所引起的偏差不超过最大允许值时的采样周期定为系统允许的最大采样周期。

(4) 采样周期对系统输出平滑性的影响

通过零阶保持器输出的微机控制指令信号是一组阶梯信号，而被控过程的输出是一组彼此相连的阶跃响应，信号阶梯随着采样周期增大，使被控对象的输出响应不平滑，产生不允许的高频波动。为了减小这种波动，应保证响应过程中有足够多的采样点数。经验规则是：

- 如果系统的阶跃响应是非周期的，则在阶跃响应的上升时间 T_r 内，采样点数 $N_r \geq (5 \sim 10)$ ；
- 如果系统的阶跃响应是振荡的，则在一个振荡周期 T_d 内，采样点数 $N_r \geq (10 \sim 20)$ 。

零阶保持器会引起相位滞后 $\Phi = \omega T/2$ ，这也是降低系统稳定性的重要原因。为了保证系统有足够的相稳定裕度，要求零阶保持器在系统开环截止频率 ω_c 附近引入的相位滞后不大于 $5^\circ \sim 10^\circ$ ，以保证对原系统的相稳定裕度影响不致过大，即要求采样周期：

$$T \leq (0.17 \sim 0.35)/\omega_c$$

(5) 采样周期与计算机字长

采样周期也不是取得越小越好，因为这样增加了微机的负担，而且即使采样周期趋于无限小。由于计算机运算部件、A/D 及 D/A 变换器的字长有限，计算机控制系统并不趋近于连续系统，反而会增大量化误差的影响。另外，还可能出现两次采样间的偏差变化太小，在几个采样周期内被采样的信号输出保持相对不变，从而对实现差分算法不利。

由于字长有限，不能准确表示控制算法参数，所以当采样周期过小时，还会增大参数变化对控制算法的影响。

(6) 采样周期对计算机负荷的影响

微机控制系统要求实时在线运行的计算机在一个采样周期内完成必要的系统管理、输入输出、控制算法计算等任务。但各项任务都要占用一定的时间，所以，采样周期的下限由微机执行控制程序和输入输出所耗费的时间决定。

尽管微型计算机的运算速度越来越快，但控制系统的算法也越来越复杂，计算工作量也限制了采样周期的降低。

2. 选择采样周期的经验规则

合理地缩短采样周期可能使多数控制系统性能得到改善，但又会加重计算负担，增加计算机的成本，到目前为止，很难找到最优采样周期的定量计算方法。下列经验规则，可以作为工程应用的参考：

- ① 如果闭环控制系统被控过程主导极点的时间常数为 T_d ，则通常取采样周期

$$T < T_d/10$$

如果被控过程的开环特性较差（即主导极点的 T_d 较大），而对闭环特性要求较高，则采样周期应取得更小。

- ② 如果被控过程具有纯延滞时间 τ ，且影响较大，则通常取采样周期

$$T < (1/4) \tau \sim (1/10) \tau$$

- ③ 如果闭环系统要求稳态调节时间为 t_s ，闭环自然频率为 ω_n ，则通常取

$$\text{采样周期 } T < t_s/10 \quad \text{或} \quad \text{采样频率 } \omega_s > 10\omega_n$$

多数工业过程控制系统常用的采样周期为几秒到几十秒，快速的机电控制系统则要求采

样周期为毫秒级或更短。常用控制对象采样周期的经验数据如表 2.1 所示。

表 2.1 常用被控对象采样周期的经验数据

控制对象	采样周期 /s
流量	1~5（优先取 1~2）
压力	3~10
液位	6~10
温度	10~20

2.2.2 常用数据处理

1. 系统误差补偿

系统误差是指在相同条件下，经过多次测量，数值大小和正负符号保持恒定的误差，或按某种已知的规律变化的误差。在微机控制系统的测量输入通道中，总存在着不同程度的零点偏移和漂移、放大电路的增益变化、器件参数的不稳定等现象，影响到测量数据的准确性，而其特点是，在一定的测量条件下，变化规律是可以掌握的，产生误差的原因一般也是知道的。因此，这些误差都属于系统误差。

通常认为，在系统的实际输入值 x 与测量输出值 y 之间存在线性关系：

$$(y-y_0)/(y_1-y_0) = x/x_1 \tag{2.5}$$

式中， y_0 是实际输入值为 0 时的测量输出值； y_1 是实际输入值为基准值 x_1 时的测量输出值。

基于这种线性关系，微机控制系统可以通过校准电路（如图 2.14 所示）和程序的配合，对系统误差加以校正。

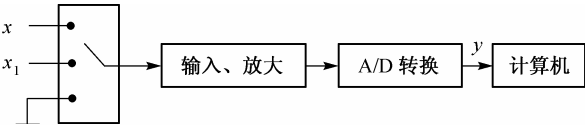


图 2.14 校准电路

系统在刚上电时或每隔一定时间进行一次校准。校准时，先把开关接地，测出这时的输出值 y_0 ，然后把开关接基准值 x_1 ，测出输出值 y_1 ，并存放 y_0 和 y_1 。这样，根据测得的输出值 y ，就可根据上述线性关系，通过程序计算得出实际的输入值 x 。

采用这种方法测得的 x ，可以补偿输入电路、放大电路、A/D 转换器本身的零点偏移及随时间、温度而发生的各种漂移变化的影响，与 x_1 的精度也无关。从而提高了测量精度，降低了对电路器件的要求和硬件成本。

如果只考虑零点偏移的校准，可以采用简单的数字调零，即将测得的输出值 y 减去输入接地时测出的输出值 y_0 ，认为 $(y-y_0)$ 才是与实际的输入值 x 相对应的。

如果还要考虑到非电物理量转换为电信号环节，即传感器引入的误差，则可以在需要时接入标准的非电物理参数进行测量，再按下式校准：

$$(y-y_0)/(y_k-y_0) = x/x_k \tag{2.6}$$

式中， y_0 是输入为 0 时的测量输出值； y_k 是输入为标准的非电物理参数 x_k 时的测量输出值。

传感器误差校正特别适用于传感器特性会随时间发生变化的场合。例如，一般的湿敏传

感器特性一年以上的变化会大于精度容许值，每隔一段时间（例如，一个月或三个月），可以用其他方法测出当时的湿度值，然后把它作为标准值输入测量系统，并记录此时的输出值。微机控制系统可以用它们来自动校准以后的测量值。

2. 非线性补偿

控制系统实际工作时，经常会遇到非线性的函数关系。简单的解析式可以采用牛顿迭代法进行计算，但有时解析式比较复杂，计算麻烦，速度也太慢，有的甚至很难写出解析式，只能根据实验观测或其他方法来确定与自变量某些值对应的函数值。微机控制系统常用的处理办法是用表格法和插值法来计算补偿值。

(1) 表格法

将 $y=f(x)$ 的函数关系以表格方式存储起来，可以方便地根据输入 x 查找到输出 y ；如果表格存储值不恰好等于输入值，可以查找到与输入 x 相邻的上下值，按线性比例关系计算出相应的 y 近似值。

(2) 插值法

对于不便于计算或处理的函数，可以建立一个简单的近似表达式。插值法是一种建立近似公式的基本方法，用这种方法得到的近似公式称为插值公式。

① 建立插值公式的原则。假定已知函数 $f(x)$ 在 $n+1$ 个相异点

$$a = x_0 < x_1 < \cdots < x_n = b$$

处的函数值为

$$f(x_0)=y_0, f(x_1)=y_1, \cdots, f(x_n)=y_n$$

如果找到某一个函数 $P(x)$ 去逼近 $f(x)$ ，并使 $P(x)$ 在 $x_i (i=0\sim n)$ 处与 $f(x_i)$ 相等，则称 $P(x)$ 为 $f(x)$ 的插值函数， x_i 为插值节点。在区间 $[a, b]$ 可以用 $P(x)$ 来近似地代替 $f(x)$ 。

多项式是最容易计算的一类函数，因而经常选择 $P(x)$ 为 n 次多项式，这种插值方法称为代数插值，也称为多项式插值，即用 $P_n(x)=a_nx^n+a_{n-1}x^{n-1}+\cdots+a_1x+a_0$ 去逼近 $f(x)$ ，使 $P_n(x)$ 在节点 x_i 处满足 $P_n(x_i)=f(x_i)=y_i, i=0, 1, \cdots, n$ 。

由于多项式 $P_n(x)$ 中的未定系数有 $n+1$ 个 (a_n, \cdots, a_0) ，而它所应满足的条件也有 $n+1$ 个，即有

$$\begin{aligned} a_nx_0^n + a_{n-1}x_0^{n-1} + \cdots + a_1x_0 + a_0 &= y_0 \\ a_nx_1^n + a_{n-1}x_1^{n-1} + \cdots + a_1x_1 + a_0 &= y_1 \\ &\cdots\cdots\cdots \\ a_nx_n^n + a_{n-1}x_n^{n-1} + \cdots + a_1x_n + a_0 &= y_n \end{aligned} \tag{2.7}$$

这是一个含 $n+1$ 个未知数 $a_n, a_{n-1}, \cdots, a_1, a_0$ 的线性方程组，在 x_0, x_1, \cdots, x_n 互异时，有唯一的一组解，因此一定存在一个唯一的 $P_n(x)$ 满足所要求的插值条件。

② 建立插值公式的方法。高斯消去法可以直接得到多项式 $P_n(x)$ 中的各项系数，比较适合微机控制系统进行实时计算的需要，因而最常用。其基本思路是，先根据已知的 x_i 和 $y_i (i=0, \cdots, n)$ 离线求解线性方程组，求出 $a_i (i=0, \cdots, n)$ 及插值多项式 $P_n(x)$ ；在实时系统中，根据输入值 x 近似地实时计算 $f(x) \approx P_n(x)$ 。

③ 误差控制。插值多项式 $P_n(x)$ 只是近似于被逼近的函数 $f(x)$ ，不能在整个 $[a, b]$ 区间都等于 $f(x)$ ，插值多项式的余项 $R(x)=|f(x)-P_n(x)|$ 实际就代表了误差，它取决于多项式次数 n 、插值节点 x_i 和被逼近函数 $f(x)$ 的特性，所以，通过选择插值多项式次数 n 、插值节点 (x_i, y_i) 和

自变量取值范围的分段，可以控制误差，保证所需的插值精度。

(3) 插值多项式的应用

在微机控制系统中，经常会遇到需要对各种传感器的输入进行非线性补偿或需要进行复杂的变换、计算等情况。这时，可采用插值多项式来完成补偿计算。

为了提高插值逼近精度，单纯地提高多项式的次数可能产生摆动现象，常用和容易实现的方法是选取适当的插值节点并分段插值。即把自变量的取值范围分成适当的几段来进行插值，这样各段多项式的次数要比整体一起插值的次数大大降低，甚至用几段折线就可以达到目的。难以进行一次性插值的函数，特别是分段函数，更需要使用分段插值方法。

【例 2.2】计算热敏电阻温度——电阻特性的插值多项式

一种热敏电阻的温度——电阻特性如表 2.2 所示，设在 10℃~40℃ 范围内使用该热敏电阻作为温度传感器，并且要求误差小于 0.05℃。

运行根据高斯消去法或拉格朗日插值法编制的程序后，先试探输入 $N=2$ ，并选择插值节点为(8.000, 10℃)、(6.1538, 25℃)、(5.000, 40℃)，得到插值多项式。然后代入其他电阻值 R ，计算出相应温度值 t ，并与表 2.2 所列值比较，发现最大误差为 0.4℃ 左右，超过要求值(0.05℃)。为此，可取 $N=3$ ，取插值节点为(8.000, 10℃)、(6.667, 20℃)、(5.7142, 30℃)、(5.000, 40℃)，重新运行该程序。这时，得到的插值多项式最大误差为 0.07℃，发生于 15℃ 左右，而其他处误差均小于 0.03℃。因此，可加密 10℃~20℃ 附近的节点，取插值节点为(8.000, 10℃)、(7.0174, 17℃)、(5.9701, 27℃)、(5.0631, 39℃)，此时最大误差约为 0.03℃。得到的多项式为

$$t = P_3(R) = -0.234\ 698\ 9R^3 + 6.120\ 273R^2 - 59.280\ 43R + 212.7118$$

表 2.2 一种热敏电阻的温度-电阻特性

温度 t /(°C)	电阻 R /kΩ	温度 t /(°C)	电阻 R /kΩ	温度 t /(°C)	电阻 R /kΩ
10	8.000	21	6.5574	31	5.6337
11	7.8431	22	6.4516	32	5.5554
12	7.6923	23	6.3491	33	5.4793
13	7.5471	24	6.250	34	5.4053
14	7.4074	25	6.1538	35	5.3332
15	7.2727	26	6.0606	36	5.263
16	7.1428	27	5.9701	37	5.1946
17	7.0174	28	5.8823	38	5.1281
18	6.8965	29	5.797	39	5.0631
19	6.7796	30	5.7142	40	5.000
20	6.667				

在微机控制系统中，热敏电阻值 R 一般由测量电路直接测出，温度 t 的补偿计算由在线程序完成。按上列多项式编制这个在线程序时，可利用递归等技巧：

$$t = a_3R^3 + a_2R^2 + a_1R + a_0 = [(a_3R + a_2)R + a_1]R + a_0 \tag{2.8}$$

2.3 数 字 滤 波

为了准确地进行测量和控制，必须消除输入信号中来自被测信号源、传感器、外界干扰的种种噪声。噪声可分为两类：周期性的噪声和不规则随机性的噪声。

对于周期性的噪声，采用积分时间等于噪声周期整数倍的双积分 A/D 转换器，能有效地消除

其影响。例如，50 Hz 工频干扰这类典型的周期性干扰，可以采用积分时间等于 20 ms 的整数倍。

用数字滤波削弱或滤除随机干扰的方法在微机控制系统中得到了广泛的应用。

数字滤波是一种软件滤波，就是通过程序计算或判断来减小噪声在有用信号中的比重。与硬件的模拟滤波器相比，数字滤波有如下优点：

- 用程序实现，不需要增加硬件设备，可靠性高，稳定性好；
- 可以对频率很低（如 0.01 Hz）的信号实现滤波，克服了模拟滤波器的缺陷；
- 模拟滤波器通常需要对每个通道都设置，而数字滤波器可以多个通道共用，从而降低成本；
- 可以根据信号的不同，采用不同的数字滤波方法或滤波参数，灵活、方便、功能强。

数字滤波器的缺点是需要程序运行时间，因而实时性较差。随着信号处理专用芯片 DSP 技术的发展和算法的改进，柔性实时处理平台的功能越来越强大，这一缺点得到了弥补。

2.3.1 常用数字滤波方法

1. 算术平均值法

可以证明，对于输入的 N 个采样数据 x_i ($i = 1, \dots, N$)

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i \quad (2.9)$$

与各采样值 x_i 间的偏差的平方和为最小。

设进行 N 次测量后，信号的平均幅度为 s 、噪声的平均幅度为 C 。第 i 次测量的测量值包括信号成分 s_i 和噪声成分 C_i ，则进行 N 次测量信号成分之和为

$$\sum_{i=1}^N s_i = N \cdot s$$

噪声的强度是用均方根来衡量的，当噪声为随机信号时，进行 N 次测量的噪声强度之和为

$$\sum_{i=1}^N C_i^2 = N \cdot C^2$$

这样，对 N 次测量进行算术平均后的信噪比为

$$\frac{N \cdot s}{\sqrt{N} \cdot C} = \sqrt{N} \cdot \frac{s}{C} \quad (2.10)$$

而求算术平均值前的信噪比是 s/C 。可见，采用算术平均值法后，信噪比提高了 \sqrt{N} 倍。

算术平均值法适用于对一般的有随机干扰的信号滤波。它特别适用于信号本身在某一数值范围附近进行上下波动的情况，如流量、液平面等信号的测量。算术平均值法对信号的平滑滤波程度完全取决于 N 。当 N 较大时，平滑度高，但灵敏度低，即外界信号的变化对测量计算结果 y 的影响小；当 N 较小时，平滑度较低，但灵敏度高。

N 应按具体情况选取，如一般流量测量，可取 $N = 8 \sim 16$ ；压力等测量，可取 $N = 4$ 。

2. 滑动平均值法

一般的算术平均值法，需测量 N 次才计算一次数据，不能用于测量速度较慢或要求数据计算速率较高的实时系统。

滑动平均值法每进行一次测量，就能得到一个新的算术平均值。

滑动平均值法可以采用长度为 N 的循环队列作为测量数据存储器。每进行一次测量，把测量结果放入队尾，而去掉原来队首的一个数据，这样在队列中始终有 N 个最新的测量数据。

对队列中的 N 个数据进行算术平均，就可以得到新的算术平均值。

算术平均的缺点是， N 越大反应越迟钝，对正确的测量值和干扰的影响也不能加以区别。

3. 加权平均值法

上述两种算术平均方法中，所有 N 次采样得到的各个值对结果的影响是相同的。在某些应用场合，为了提高滤波效果，可以采用加权平均值法。

N 项加权平均式为

$$\bar{Y} = \sum_{i=1}^N C_i x_i \quad (2.11)$$

式中， C_1, C_2, \dots, C_N 是各次采样值的加权系数，应满足 $\sum_{i=1}^N C_i = 1$ 。

加权系数可根据具体情况决定，目的是增加信号的一部分，抑制信号的另一部分。一般采样次数越靠后，取的比例越大，可以增加新的采样值在平均值中的比例。

4. 防脉冲干扰平均值法

在微机控制系统中，经常会遇到尖脉冲干扰的现象。干扰通常只影响个别采样点的数据，此数据与其他采样点的数据相差比较大。如果采用一般的平均值法，则干扰也将平均到计算结果上去，引起的偏差不易被消除。为此，可以采用类似于体操等比赛评分的方法，即先对 N 个数据进行比较，去掉其中最大值和最小值，然后计算余下的 $N-2$ 个数据的算术平均值。这种方法既可滤去脉冲干扰，又可滤去小的随机干扰。

为了加快测量计算速度，一般取 $N=4$ 或 6 ，然后进行 4 取 2 或 6 取 4 ，再平均。这样，计算方便，速度快、占用存储容量小。

5. 程序判断滤波法

程序判断滤波可以有效地滤去由于随机干扰、误检测或者转换器不稳定而引起的信号严重失真。根据经验，确定采样输入信号的上下限幅值或两次采样输入信号可能出现的最大偏差。若采样数据偏离太远，则认为本次输入信号无效。按判断的依据，可分为限幅滤波和限速滤波两种。

(1) 限幅滤波

限幅滤波限制的是采样输入信号的上下限幅值，若取 y_H 为上限值， y_L 为下限值，则

当 $y(n) \geq y_H$ 时，取 $y(n) = y_H$ ；

当 $y(n) \leq y_L$ 时，取 $y(n) = y_L$ ；

当 $y_L < y(n) < y_H$ 时，取 $y(n)$ 。

(2) 限速滤波

限速滤波限制的是两次相邻采样值的变化量。若取 Δy_0 为两次相邻采样值之差的最大容许值，则

当 $|y(n) - y(n-1)| \leq \Delta y_0$ 时，取 $y(n)$ ；

当 $|y(n) - y(n-1)| > \Delta y_0$ 时，取 $y(n) = y(n-1)$ 。

Δy_0 取决于采样周期 T 及被测参数 y 应有的正常变化率。为了保证滤波效果和控制品质，要按照实际情况来确定 $\Delta y_0, y_L$ 和 y_H 。

6. 惯性滤波法

常用的 RC 低通滤波器的输入电压 $x(t)$ 和输出电压 $y(t)$ 的关系是

$$RCdy(t)/dt + y(t) = x(t) \quad (2.12)$$

RC 滤波器不可能对极低频率的信号进行滤波。低通特性的一阶惯性数字滤波器可以比模拟滤波器有更好的滤波特性。

将微分公式写成差分方程

$$RC(y(n) - y(n-1)) / \Delta t + y(n) = x(n) \quad (2.13)$$

稍加整理后, 得

$$y(n) = ay(n-1) + bx(n) \quad (2.14)$$

其中, a 称为滤波系数, $b = 1 - a$ 。

根据惯性滤波器的频率特性, 若滤波系数 a 越大, 则带宽越窄, 滤波频率也越低。需要根据实际情况, 适当选取 a 值, 使得被测参数既不会出现明显的纹波, 反应又不至于迟缓。

2.3.2 数字滤波方法的选择

数字滤波方法和参数可以根据检测对象、干扰性质的不同及对实时性的要求进行选择。

1. 根据对象选择

- 对于变化比较缓慢的参数如温度, 可以选择程序判断滤波和惯性滤波;
- 对于变化比较快的参数如压力、流量, 可以选择算术平均、加权平均、防干扰平均滤波;
- 对于纯迟延较大的被控对象, 可以选择加权平均滤波。

采样次数 N 越大, 滤波效果越好, 但花费的时间则越长。

2. 根据干扰性质选择

- 平均值滤波法适用于周期性干扰;
- 加权平均滤波法、限幅滤波法和防脉冲干扰平均值法适用于偶然的脉冲干扰;
- 惯性滤波法适用于高频及低频的干扰信号。

如果滤波效果要求更高, 可以采用几种方法复合滤波, 例如, 先用程序判断, 再用平均滤波。

应该注意的是, 当设计微机控制系统时, 是否需要数字滤波, 采用哪一种滤波方法, 都只有经过实验调试才能最后确定, 采用不适当的数字滤波可能适得其反。

习题与思考题

- 1 堆栈和队列各适用于什么场合? 试各举一例说明。
- 2 参考图 2.4 和图 2.5, 编写加入和读出循环队列的程序。
- 3 参考图 2.9, 编写将链表的结点 K 删除后放入自由表中的操作步骤的流程图和程序。
- 4 如果给定数据可能位于表中数值之间, 此时应如何编制查表程序。
- 5 从空间效率和时间效率两个角度来分析, 顺序表和链表分别更适用于哪些场合?
- 6 试设计一个从一组数据中快速找出最大值的子程序。
- 7 一种热敏电阻的温度-电阻特性如表 2.2 所示, 试用分段法编制热敏电阻补偿计算程序。
- 8 数字滤波有什么优点和缺点? 应该怎样来选用各种数字滤波方法。
- 9 试编制一个对 N 个采样值有防脉冲干扰功能的滑动平均值数字滤波子程序。
- 10 试编制具有高通特性和带通特性的数字滤波子程序。

第 3 章 微机控制系统与信号的数学描述

3.1 微机控制系统中的信号变换

如图3.1所示，微型计算机控制系统是模拟部件与数字部件共存的混合系统，进行着多种信号（见表3.1）的传输和变换。从时间上看，既有时间轴上任何时刻都存在的连续信号，也有时间轴上断续出现的离散信号；从幅值上看，既有幅值连续变化，可取任意值的模拟量，也有幅值上只能取离散值的离散量及数字量（用一定位数的二进制编码表示的离散量）。

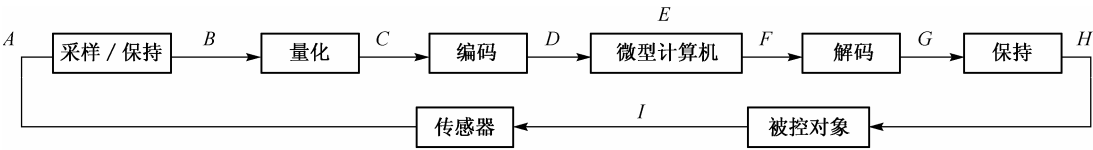


图 3.1 微机控制系统中的信号变换

表 3.1 微机控制系统中的多种信号

信号形式		图形表示	图 3.1 中各点信号
时 间	幅 值		
连续	模拟量		A, I
连续	阶梯形模拟量		H
离散	模拟量		B
离散	离散量		C, G
离散	数字量		D, F
连续	数字量		E （计算机内存信号）

3.1.1 信号的采样与保持

在微机控制系统中,除了直接接收和发送数字信号以外,还需要通过 A/D 转换,按照采样/保持、量化和编码的顺序,将连续模拟信号变换为数字计算机能接收和处理的离散数字编码信号;另一方面,为了控制连续式器件,又需要解码器与保持器组合的 D/A 转换,将数字编码信号转换为相应的时间连续的电流或电压阶梯状模拟信号。

把连续信号转变为离散脉冲序列的过程称为采样。实现采样的过程,可以用一个周期性闭合的采样开关 S 来表示(如图 3.2(a)所示)。

用 T 表示采样周期,单位为 s ; $f_s = 1/T$ 表示采样频率,单位为 $1/s$; $\omega_s = 2\pi f_s = 2\pi/T$ 表示采样角频率,单位为 rad/s 。在实际应用中,采样开关多为电子开关,闭合时间极短,采样持续时间 τ 远小于采样周期 T ,也远小于系统连续部分的最大时间常数。为了简化系统的分析,可认为 τ 趋于零,即把采样器的输出近似看成一串强度等于矩形脉冲面积的理想脉冲 $e^*(t)$,如图 3.2(c)所示。

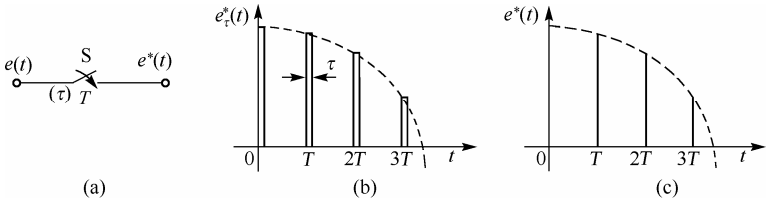


图 3.2 采样器的输入与输出信号

把脉冲序列转变为连续信号的过程称为信号的复现。实现复现过程的装置称为保持器。采用保持器不仅是实现两种信号之间转换的需要,也是因为采样器输出的是脉冲信号 $e^*(t)$,如果不经滤波将其恢复成连续信号,则 $e^*(t)$ 中的高频分量相当于给系统中的连续部分加入了噪声,不但影响控制质量,严重时还会加剧部件的磨损。因此,需要在采样器后面串联一个信号复现滤波器,使脉冲信号 $e^*(t)$ 复原成连续信号,再加入到系统的连续部分。最简单的复现滤波器由零阶保持器实现,可把脉冲信号 $e^*(t)$ 复现为阶梯信号 $e_h(t)$,如图 3.3 所示。当采样频率足够高时, $e_h(t)$ 接近于连续信号。

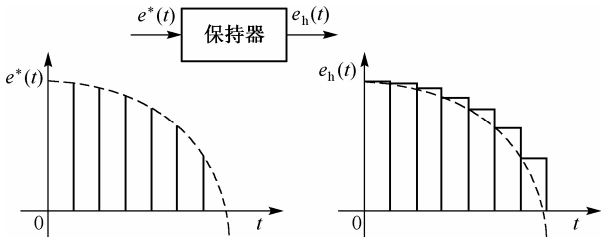


图 3.3 保持器的输入与输出信号

3.1.2 采样定理

采样过程在时域内只保留了采样时刻的信息,丢失了采样间隔之间的信息。如果采样频率足够高,采到的信号很密,则采样信号就可以近似代表原来的连续信号。采样定理回答了需要满足什么条件,采样信号才可以不失真地代表原来的连续信号:

若连续信号是有限带宽的，且它所含的频率分量的最大值为 ω_m ，则当采样频率 $\omega_s \geq 2\omega_m$ 时，原连续信号完全可以用其采样信号来表征，或者说采样信号可以不失真地代表原连续信号。

也可以用采样周期来表述采样定理：若连续信号的最高频率为 ω_m ，则均匀采样周期 $T \leq \pi/\omega_m$ 。其物理意义是，如果对于连续信号中所含的最高频率的正弦分量，在 1 个周期内采样 2 次以上，那么经采样得到的脉冲序列，就包含了连续信号的全部信息；如果采样次数太少，采样信号就不可能不失真地反映连续信号的特性。

如图 3.4 所示，频率为 $f_1 = 7/8 \text{ Hz}$ 的余弦信号，当用采样频率 $f_s = 1 \text{ Hz}$ 进行采样时，由于不满足采样定理要求，采样间隔内的信息丢失太多，采样的结果看起来就像一个 $f_s - f_1 = 1/8 \text{ Hz}$ 的低频信号。这种现象被称为假频现象。

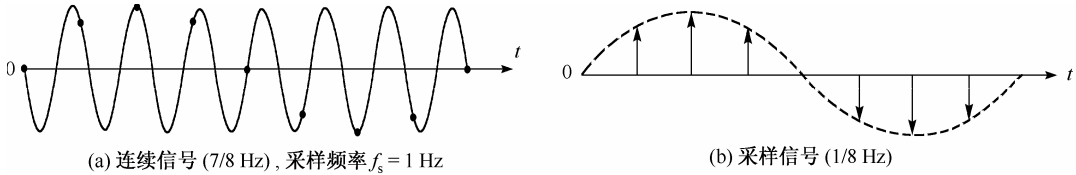


图 3.4 假频现象

采样定理给出的是由采样脉冲序列无失真地再现原连续信号所允许的最低采样频率。在控制工程实践中，采样周期的选择一般总是取 $\omega_s > 2\omega_m$ ，如前 2.2.1 节所述。

3.1.3 离散时间信号

微机控制系统中，对连续信号采样，就可得到离散时间信号。

1. 定义

离散系统中，用离散时间的数字序列来表示信号。序列中的第 n 个数字记为 $x(n)$ ，全部信号序列可写为

$$x = \{x(n)\} \quad -\infty < n < +\infty \quad (3.1)$$

$x(n)$ 仅对于整数 n 值才有定义，对于非整数的 n 值无定义。

2. 常用序列

(1) 单位采样序列（单位脉冲序列）

$$\delta(n) = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases} \quad (3.2)$$

(2) 单位阶跃序列

$$u(n) = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases} \quad (3.3)$$

(3) 单位采样序列和阶跃序列的关系

$$u(n) = \sum_{k=0}^{\infty} \delta(n-k) \quad (3.4)$$

$$\delta(n) = u(n) - u(n-1) \quad (3.5)$$

3. 离散信号的运算

可以用离散时间信号序列定义线性离散系统的下列运算：

$$x \cdot y = \{x(n) \cdot y(n)\}$$

$$x + y = \{x(n) + y(n)\}$$

$$ax = \{ax(n)\} \quad (3.6)$$

序列 x 延迟或平移一个序数 n_0 , 称为位移序列

$$y(n) = x(n - n_0) \quad (3.7)$$

单位采样序列的位移序列为

$$\delta(n - k) = \begin{cases} 1 & n = k \\ 0 & n \neq k \end{cases} \quad (3.8)$$

任何序列 $x(n)$ 都可用单位采样序列表示为

$$x(n) = \sum_{k=-\infty}^{\infty} x(k) \cdot \delta(n - k) \quad (3.9)$$

连续函数 $f(t)$ 是 t 的函数, 对其采样函数 $f^*(nT)$ 来说, $t = nT$ (T 为采样周期)。一般以采样周期为单位时间, 则采样函数的变量可直接用 n 表示, 即 $f^*(n) = f^*(nT)$ 。

连续函数 $f(t)$ 的采样函数 $f^*(n)$ 也就是它的离散信号序列, 为

$$f^*(n) = \sum_{k=0}^{\infty} f(k) \cdot \delta(n - k) \quad (3.10)$$

3.2 线性定常离散系统的描述与差分方程

在微机控制系统中, 将不同形式的信号都转换为时间离散信号, 一般更为方便。应用差分方程、 Z 变换、脉冲传递函数等数学工具, 可以将微机控制系统作为线性定常离散系统进行描述和研究。

3.2.1 线性定常离散系统的描述

在数学上, 线性离散系统可定义为将输入 $x(n)$ 映射为输出 $y(n)$ 的唯一性变换, 即

$$y(n) = T[x(n)] \quad (3.11)$$

线性离散系统具有叠加性:

$$T[ax_1(n) + bx_2(n)] = aT[x_1(n)] + bT[x_2(n)] = ay_1(n) + by_2(n) \quad (3.12)$$

输入与输出关系不随时间而改变的线性离散系统, 称为线性定常离散系统, 具有位移不变性。

若 $y(n) = T[x(n)]$, 则

$$y(n - n_0) = T[x(n - n_0)] \quad (3.13)$$

$$\begin{aligned} y(n) &= T[x(n)] = T\left[\sum_{k=-\infty}^{\infty} x(k)\delta(n - k)\right] = \sum_{k=-\infty}^{\infty} x(k)T[\delta(n - k)] \\ &= \sum_{k=-\infty}^{\infty} x(k)h(n - k) = x(n) * h(n) \end{aligned} \quad (3.14)$$

式中, $h(n - k) = T[\delta(n - k)]$, 表示系统对单位采样序列的响应。

$x(n) * h(n)$ 为卷积运算, 定义为

$$x(n) * h(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k) \quad (3.15)$$

对于任何一个有界输入, 其输出都是有界的时域离散系统称为稳定系统。稳定系统的充要条件为

$$\sum_{k=-\infty}^{\infty} |h(k)| < \infty \quad (3.16)$$

3.2.2 线性常系数差分方程

时域定常线性连续系统可用如下微分方程表示:

$$\sum_{k=0}^n a_k \frac{d^k y(t)}{dt^k} = \sum_{r=0}^m b_r \frac{d^r x(t)}{dt^r} \quad (3.17)$$

设采样间隔为 T , 则有

一阶后向差分:

$$y'(n) \approx \frac{y(n) - y(n-1)}{T}$$

或一阶前向差分:

$$y'(n) \approx \frac{y(n+1) - y(n)}{T}$$

二阶差分:

$$y''(n) \approx \frac{y(n) - 2y(n-1) + y(n-2)}{T^2}$$

依次类推, 时域定常线性离散系统可用如下线性常系数差分方程表示:

$$\sum_{k=0}^n C_k y(n-k) = \sum_{r=0}^m d_r x(m-r) \quad (3.18)$$

当给定起始条件时, 在由差分方程表示的系统中, 输出的第 n 个值可由第 m 个输入值及输出的前 n 个过去值和输入的前 m 个过去值计算求得

$$y(n) = -\sum_{k=1}^n \frac{C_k}{C_0} y(n-k) + \sum_{r=0}^m \frac{d_r}{C_0} x(m-r) \quad (3.19)$$

差分方程在数学上表示了系统, 还可以为系统的计算机实现提供基础: 根据差分方程, 就可以编制程序流程图和程序。

常系数线性差分方程的求解方法有经典法、迭代法和 Z 变换法。

与微分方程的经典解法类似, 差分方程的经典解法也需要求出齐次方程的通解和非齐次方程的一个特解。工程上常用后两种解法。

3.2.3 应用迭代法求解差分方程

若已知差分方程式(3.18)或式(3.19), 并且给定输出序列的初值, 则可以利用递推关系, 逐步算出输出序列。

【例 3.1】 已知差分方程

$$c(k) = r(k) + 5c(k-1) - 6c(k-2)$$

输入序列 $r(k)=1$ ，初始条件为 $c(0)=0, c(1)=1$ ，试用迭代法求输出序列 $c(k)$ ， $k=0,1,2,\cdots,10$ 。

解：根据初始条件及递推关系，得

$$\begin{aligned} c(0) &= 0 \\ c(1) &= 1 \\ c(2) &= r(2) + 5c(1) - 6c(0) = 6 \\ c(3) &= r(3) + 5c(2) - 6c(1) = 25 \\ c(4) &= r(4) + 5c(3) - 6c(2) = 90 \\ c(5) &= r(5) + 5c(4) - 6c(3) = 301 \\ c(6) &= r(6) + 5c(5) - 6c(4) = 966 \\ c(7) &= r(7) + 5c(6) - 6c(5) = 3025 \\ c(8) &= r(8) + 5c(7) - 6c(6) = 9330 \\ c(9) &= r(9) + 5c(8) - 6c(7) = 28\,501 \\ c(10) &= r(10) + 5c(9) - 6c(8) = 86\,526 \end{aligned}$$

3.3 Z变换及其应用

像连续系统引入拉氏变换一样，对离散系统引入 Z 变换，可把差分方程简化为代数方程，从而方便求解。还可以得到描述离散系统的脉冲传递函数，方便地分析线性离散系统的稳定性、稳态特性和动态特性，使系统的分析研究大大简化。因此， Z 变换是分析、研究和设计线性离散系统的重要工具。

3.3.1 Z变换定义

离散系统 $x(n)$ 的 Z 变换定义为

$$X(z) = \sum_{k=0}^{\infty} x(kT)z^{-k} = x(0) + x(T)z^{-1} + x(2T)z^{-2} + x(3T)z^{-3} + \cdots \quad (3.20)$$

式中， $z = e^{\sigma + j\omega} = re^{j\omega}$ ， $r = |z| = e^{\sigma}$ ， k 为整数， T 为采样周期。

Z 变换收敛的充要条件是

$$\sum_{n=0}^{\infty} |x(n)r^{-n}| < \infty \quad (3.21)$$

一般情况下，系统在 $n < 0$ 时， $x(n) = 0$ ，故单边 Z 变换和双边 Z 变换是等同的。

为了方便，对 $f(t)$ 进行 Z 变换，实际上都是对其采样函数 $f^*(t)$ 进行的 Z 变换，因为连续函数 $f(t)$ 不能直接进行 Z 变换，只有采样函数才能定义 Z 变换。

对 于

$$x^*(n) = \sum_{k=0}^{\infty} x(t) \cdot \delta(nT - kT)$$

(3 . 2 2)
求拉氏交换，得

$$X^*(s) = L[x^*(t)] = \int_0^{\infty} x^*(t) e^{-st} dt = \int_0^{\infty} \sum_{k=0}^{\infty} x(t) \delta(nT - kT) \cdot e^{-st} dt = \sum_{k=0}^{\infty} x(kT) e^{-kTs} \quad (3.23)$$

与 Z 变换定义相对比, 可以看出, Z 变换的复变量 z 与拉氏变换的复变量 s 之间存在关系:

$$z = e^{Ts} \quad (3.24)$$

可以将 Z 变换看做一种在采样拉氏变换中, 将 s 的超越函数转换为 z 的幂级数或 z 的有理分式的变量置换。

如果不相同的连续信号 $f_1(t), f_2(t), \dots$, 它们的采样序列 $f^*(k)$ 相同, 则其 Z 变换式相同。也就是说, 一个像函数 $F(z)$ 可能有多个原函数 $f(t)$, 它们在 $t = kT$ ($k=0, 1, 2, \dots$) 的诸时刻上的值 $f(kT)$ 相等。

3.3.2 Z 变换方法

在求离散时间函数的 Z 变换方法中, 级数求和法和部分分式法比较常用。

1. 级数求和法

根据给定的理想采样开关的输入连续信号 $e(t)$ 或其输出采样信号 $e^*(t)$, 以及采样周期 T , 由式(3.20)可得 Z 变换级数展开式。通常, 对于常用函数 Z 变换的级数形式, 都可以写出其闭合形式。

【例 3.2】 求单位阶跃函数 $u(t)$ 的 Z 变换。

解: 由于 $e(t) = u(t)$ 在所有采样时刻上的采样值均为 1, 即 $e(nT) = 1$ ($n=0, 1, 2, \dots, \infty$), 故由式(3.20), 有

$$E(z) = 1 + z^{-1} + z^{-2} + \dots + z^{-n} + \dots$$

在上式中, 若 $|z^{-1}| < 1$, 则无穷级数是收敛的, 利用等比级数求和公式, 可得 $u(t)$ 的 Z 变换的闭合形式为

$$E(z) = \frac{1}{1 - z^{-1}} = \frac{z}{z - 1}$$

因为 $|z^{-1}| = |e^{-sT}| = e^{-\sigma T}$, 式中 $\sigma = \text{Re } s$, 所以条件 $|z^{-1}| < 1$ 意味着条件 $\sigma > 0$ 。这也是单位阶跃函数可进行拉氏变换的条件。

【例 3.3】 设

$$e(t) = \delta_T(t) = \sum_{n=0}^{\infty} \delta(t - nT)$$

试求理想脉冲序列 $\delta_T(t)$ 的 Z 变换。

解: 因为 T 为采样周期, 故

$$e^*(t) = \delta_T(t) = \sum_{n=0}^{\infty} \delta(t - nT)$$

由拉氏变换可知

$$E^*(s) = \sum_{n=0}^{\infty} e^{-nsT}$$

因此

$$E(z) = \sum_{n=0}^{\infty} z^{-n} = 1 + z^{-1} + z^{-2} + \dots$$

将上式写成闭合形式, 得 $\delta_T(t)$ 的 Z 变换为

$$E(z) = \frac{1}{1-z^{-1}} = \frac{z}{z-1}, \quad |z^{-1}| < 1$$

例 3.2 和例 3.3 的原函数不同, 但只要采样函数 $e^*(t)$ 相同, 就对应于相同的 Z 变换 $E(z)$, 这是利用 Z 变换法分析离散系统时特别需要注意的一个问题。

2. 部分分式法

利用部分分式法求 Z 变换时, 先求出已知连续时间函数 $e(t)$ 的拉氏变换 $E(s)$, 然后将有理分式函数 $E(s)$ 展成部分分式之和的形式, 使每一部分分式对应简单的时间函数, 其相应的 Z 变换是已知的, 再求出 $E(s)$ 对应的 Z 变换 $E(z)$ 。

【例 3.4】 已知连续函数的拉氏变换为

$$E(s) = \frac{a}{s(s+a)}$$

求相应的 Z 变换 $E(z)$ 。

解: 将 $E(s)$ 展成如下部分分式:

$$E(s) = \frac{1}{s} - \frac{1}{s+a}$$

对上式逐项取拉氏逆变换, 可得

$$e(t) = 1 - e^{-at}$$

利用等比级数求和公式, 可得

$$\begin{aligned} Z[1(t)] &= \frac{z}{z-1}, & Z[e^{-at}] &= \frac{z}{z-e^{-aT}} \\ E(z) &= \frac{z}{z-1} - \frac{z}{z-e^{-aT}} = \frac{z(1-e^{-aT})}{z^2 - (1+e^{-aT})z + e^{-aT}} \end{aligned}$$

【例 3.5】 设 $e(t) = \sin \omega t$, 求相应的 Z 变换 $E(z)$ 。

解: 对 $e(t) = \sin \omega t$ 取拉氏变换, 得

$$E(s) = \frac{\omega}{s^2 + \omega^2}$$

将上式展开为部分分式

$$E(s) = \frac{1}{2j} \left(\frac{1}{s-j\omega} - \frac{1}{s+j\omega} \right)$$

根据指数函数的 Z 变换表达式, 可以得到

$$E(z) = \frac{1}{2j} \left(\frac{z}{z-e^{j\omega T}} - \frac{z}{z-e^{-j\omega T}} \right) = \frac{1}{2j} \left[\frac{z(e^{j\omega T} - e^{-j\omega T})}{z^2 - z(e^{j\omega T} + e^{-j\omega T}) + 1} \right]$$

化简后, 得

$$E(z) = \frac{z \sin \omega T}{z^2 - 2z \cos \omega T + 1}$$

常用的 Z 变换公式如表 3.2 所示, 这些函数的 Z 变换都是 z 的有理分式, 且分母多项式的次数大于或等于分子多项式的次数。在表中各 Z 变换有理分式中, 分母 z 多项式的最高次数与相应传递分母 s 多项式的最高次数相等。

表 3.2 Z 变换公式

序号	拉氏变换 $E(s)$	时间函数 $e(t)$	Z变换 $E(z)$
1	1	$\delta(t)$	1
2	e^{-nsT}	$\delta(t-nT)$	z^{-n}

(续表)

3	$\frac{1}{s}$	$1(t)$	$\frac{z}{z-1}$
4	$\frac{1}{s^2}$	t	$\frac{Tz}{(z-1)^2}$
5	$\frac{1}{s^3}$	$\frac{t^2}{2!}$	$\frac{T^2z(z+1)}{2(z-1)^3}$
6	$\frac{1}{s^4}$	$\frac{t^3}{3!}$	$\frac{T^3z(z^2+4z+1)}{6(z-1)^4}$
7	$\frac{1}{s-(1/T)\ln a}$	$a^{t/T}$	$\frac{z}{z-a}$
8	$\frac{1}{s+a}$	e^{-at}	$\frac{z}{z-e^{-aT}}$
9	$\frac{1}{(s+a)^2}$	te^{-at}	$\frac{Tze^{-aT}}{(z-e^{-aT})^2}$
10	$\frac{1}{(s+a)^3}$	$\frac{1}{2}t^2e^{-at}$	$\frac{T^2ze^{-aT}}{2(z-e^{-aT})^2} + \frac{T^2ze^{-2aT}}{(z-e^{-aT})^3}$
11	$\frac{a}{s(s+a)}$	$1-e^{-at}$	$\frac{(1-e^{-aT})z}{(z-1)(z-e^{-aT})}$
12	$\frac{a}{s^2(s+a)}$	$t-\frac{1}{a}(1-e^{-aT})$	$\frac{Tz}{(z-1)^2} - \frac{(1-e^{-aT})z}{a(z-1)(z-e^{-aT})}$
13	$\frac{1}{(s+a)(s+b)(s+c)}$	$\frac{e^{-at}}{(b-a)(c-a)} + \frac{e^{-bt}}{(a-b)(c-b)} + \frac{e^{-ct}}{(a-c)(b-c)}$	$\frac{z}{(b-a)(c-a)(z-e^{-aT})} + \frac{z}{(a-b)(c-b)(z-e^{-bT})} + \frac{z}{(a-c)(b-c)(z-e^{-cT})}$
14	$\frac{s+d}{(s+a)(s+b)(s+c)}$	$\frac{(d-a)}{(b-a)(c-a)}e^{-at} + \frac{(d-b)}{(a-b)(c-b)}e^{-bt} + \frac{(d-c)}{(a-c)(b-c)}e^{-ct}$	$\frac{(d-a)z}{(b-a)(c-a)(z-e^{-aT})} + \frac{(d-b)z}{(a-b)(c-b)(z-e^{-bT})} + \frac{(d-c)z}{(a-c)(b-c)(z-e^{-cT})}$
15	$\frac{abc}{s(s+a)(s+b)(s+c)}$	$1-\frac{bc}{(b-a)(c-a)}e^{-at}-\frac{ca}{(c-b)(a-b)}e^{-bt}-\frac{ab}{(a-c)(b-c)}e^{-ct}$	$\frac{z}{z-1}-\frac{bcz}{(b-a)(c-a)(z-e^{-aT})}-\frac{caz}{(c-b)(a-b)(z-e^{-bT})}-\frac{abz}{(a-c)(b-c)(z-e^{-cT})}$
16	$\frac{\omega}{s^2+\omega^2}$	$\sin \omega t$	$\frac{z \sin \omega T}{z^2-2z \cos \omega T+1}$
17	$\frac{s}{s^2+\omega^2}$	$\cos \omega t$	$\frac{z(z-\cos \omega T)}{z^2-2z \cos \omega T+1}$
18	$\frac{\omega}{s^2-\omega^2}$	$\sinh \omega t$	$\frac{z \sinh \omega T}{z^2-2z \cosh \omega T+1}$
19	$\frac{\omega}{s^2-\omega^2}$	$\cosh \omega t$	$\frac{z(z-\cosh \omega T)}{z^2-2z \cosh \omega T+1}$
20	$\frac{\omega^2}{s(s^2-\omega^2)}$	$1-\cos \omega t$	$\frac{z}{z-1}-\frac{z(z-\cosh \omega T)}{z^2-2z \cosh \omega T+1}$
21	$\frac{\omega}{(s+a)^2+\omega^2}$	$e^{-aT} \sin \omega T$	$\frac{ze^{-aT} \sin \omega T}{z^2-2ze^{-aT} \cos \omega T+e^{-2aT}}$
22	$\frac{s+a}{(s+a)^2+\omega^2}$	$e^{-aT} \cos \omega T$	$\frac{z^2+ze^{-aT} \cos \omega T}{z^2-2ze^{-aT} \cos \omega T+e^{-2aT}}$

$$\frac{abTz}{(z-1)^2}-\frac{(a+b)z}{z-1}-\frac{b^2z}{(a-b)(z-e^{-aT})}+\frac{a^2z}{(a-b)(z-e^{-bT})}$$

23	$\frac{b-a}{(s+a)+(s+b)}$	$e^{-at} - e^{-bt}$	$\frac{z}{z-e^{-aT}} - \frac{z}{z-e^{-bT}}$
24	$\frac{a^2b^2}{s^2(s+a)(s+b)}$	$abt + (a+b) - \frac{b^2}{a-b}e^{-aT} + \frac{a^2}{a+b}e^{-bT}$	

3. Z 变换的性质和定理

Z 变换有一些基本定理与拉氏变换的基本定理相似，可以使 Z 变换的应用变得简单方便。

(1) 线性定理

若 $E_1(z) = Z[e_1(t)]$, $E_2(z) = Z[e_2(t)]$, $E(z) = Z[e(t)]$, a 为常数，则

$$Z[e_1(t) \pm e_2(t)] = E_1(z) \pm E_2(z) \quad (3.25)$$

$$Z[ae(t)] = aE(z) \quad (3.26)$$

证明：由 Z 变换定义

$$Z[e_1(t) \pm e_2(t)] = \sum_{n=0}^{\infty} [e_1(nT) \pm e_2(nT)] z^{-n} = \sum_{n=0}^{\infty} e_1(nT) z^{-n} \pm \sum_{n=0}^{\infty} e_2(nT) z^{-n} = E_1(z) \pm E_2(z)$$

以及
$$Z[ae(t)] = a \sum_{n=0}^{\infty} e(nT) z^{-n} = aE(z)$$

式(3.25)和式(3.26)表明，Z 变换是一种线性变换，其变换过程满足齐次性与均匀性。

(2) 平移定理

平移定理又称为实数位移定理，即整个采样序列在时间轴上左右平移若干采样周期，其中向左平移为超前，向右平移为滞后，如果函数 $e(t)$ 是可拉氏变换的，其 Z 变换为 $E(z)$ ，则有

滞后定理：
$$Z[e(t - kT)] = z^{-k} E(z) \quad (3.27)$$

超前定理：
$$Z[e(t + kT)] = z^k \left[E(z) - \sum_{n=0}^{k-1} e(nT) z^{-n} \right] \quad (3.28)$$

式中， k 为正整数。

算子 z 有明确的物理意义。 z^{-k} 代表时域中的滞后环节，它将采样信号滞后 k 个采样周期。 z^k 代表超前环节，它把采样信号超前 k 个采样周期。但是， z^k 仅用于运算，在物理系统中并不存在。

证明：由 Z 变换定义

$$Z[e(t - kT)] = \sum_{n=0}^{\infty} e(nT - kT) z^{-n} = z^{-k} \sum_{n=0}^{\infty} e[(n - k)T] z^{-(n-k)}$$

令 $m = n - k$ ，则有

$$Z[e(t - kT)] = z^{-k} \sum_{m=-k}^{\infty} e(mT) z^{-m}$$

由于 Z 变换的单边性，当 $m < 0$ 时，有 $e(mT) = 0$ ，所以上式可写为

$$Z[e(t - kT)] = z^{-k} \sum_{m=0}^{\infty} e(mT) z^{-m}$$

再令 $m = n$ ，证得式(3.27)。

取 $k=1$, 得

$$Z[e(t+T)] = \sum_{n=0}^{\infty} e(nT+T)z^{-n} = z \sum_{n=0}^{\infty} e[(n+1)T]z^{-(n+1)}$$

令 $m=n+1$, 上式可写为

$$Z[e(t+T)] = z \sum_{m=1}^{\infty} e(mT)z^{-m} = z \left[\sum_{m=0}^{\infty} e(mT)z^{-m} - e(0) \right] = z[E(z) - e(0)]$$

取 $k=2$, 同理得

$$\begin{aligned} Z[e(t+2T)] &= z^2 \sum_{m=2}^{\infty} e(mT)z^{-m} = z^2 \left[\sum_{m=0}^{\infty} e(mT)z^{-m} - e(0) - z^{-1}e(T) \right] \\ &= z^2 \left[E(z) - \sum_{n=0}^1 e(nT)z^{-n} \right] \end{aligned}$$

取 $k=k$ 时, 则有

$$Z[e(t+kT)] = z^k \left[E(z) - \sum_{n=0}^{k-1} e(nT)z^{-n} \right]$$

实数位移定理是一个重要定理, 其作用相当于拉氏变换中的微分和积分定理。应用实数位移定理, 可将描述离散系统的差分方程转换为 z 域的代数方程。

【例 3.6】用实数位移定理计算滞后一个采样周期的指数函数 $e^{-a(t-T)}$ 的 Z 变换, 其中 a 为常数。

解: 根据式(3.27)

$$Z[e^{-a(t-T)}] = z^{-1}Z[e^{-at}] = z^{-1} \cdot \frac{z}{z - e^{-aT}} = \frac{1}{z - e^{-aT}}$$

(3) 复数位移定理

如果函数 $e(t)$ 是可拉氏变换的, 其 Z 变换为 $E(z)$, 则有

$$Z[e^{\mp at}e(t)] = E(ze^{\pm aT}) \quad (3.29)$$

证明: 由 Z 变换定义

$$Z[e^{\mp at}e(t)] = \sum_{n=0}^{\infty} e^{\mp anT}e(nT)z^{-n} = \sum_{n=0}^{\infty} e(nT)(ze^{\pm aT})^{-n}$$

令

$$z_1 = ze^{\pm aT}$$

则有

$$Z[e^{\mp at}e(t)] = \sum_{n=0}^{\infty} e(nT)z_1^{-n} = E(ze^{\pm aT})$$

复数位移定理是仿照拉氏变换的复数位移定理导出的, 其含义是函数 $e^*(t)$ 乘以指数序列 $e^{\mp anT}$ 的 Z 变换, 就等于在 $e^*(t)$ 的 Z 变换表达式 $E(z)$ 中, 以 $ze^{\pm aT}$ 取代原算子 z 。

【例 3.7】试用复数位移定理计算函数 te^{-aT} 的 Z 变换。

解: 令 $e(t) = t$, 由表 3.2 知

$$E(z) = Z[t] = \frac{Tz}{(z-1)^2}$$

根据复数位移定理式(3.29), 有

$$E(ze^{aT}) = Z[te^{-at}] = \frac{T(ze^{aT})}{(ze^{aT} - 1)^2} = \frac{Tze^{-aT}}{(z - e^{-aT})^2}$$

(4) 终值定理

如果函数 $e(t)$ 的 Z 变换为 $E(z)$, 函数序列 $e(nT)$ 为有限值 ($n = 0, 1, 2, \dots$), 且极限 $\lim_{n \rightarrow \infty} e(nT)$ 存在, 则函数序列的终值

$$\lim_{n \rightarrow \infty} e(nT) = \lim_{z \rightarrow 1} (z-1)E(z) \quad (3.30)$$

证明: 由 Z 变换线性定理, 有

$$Z[e(t+T)] - Z[e(t)] = \sum_{n=0}^{\infty} \{e[(n+1)T] - e(nT)\} z^{-n}$$

由平移定理

$$Z[e(t+T)] = zE(z) - ze(0)$$

于是

$$(z-1)E(z) - ze(0) = \sum_{n=0}^{\infty} \{e[(n+1)T] - e(nT)\} z^{-n}$$

上式两边取 $z \rightarrow 1$ 时的极限, 得

$$\begin{aligned} \lim_{z \rightarrow 1} (z-1)E(z) - e(0) &= \lim_{z \rightarrow 1} \sum_{n=0}^{\infty} \{e[(n+1)T] - e(nT)\} z^{-n} \\ &= \sum_{n=0}^{\infty} \{e[(n+1)T] - e(nT)\} \end{aligned}$$

当取 $n = N$ 为有限项时, 上式右端可写为

$$\sum_{n=0}^{\infty} \{e[(n+1)T] - e(nT)\} = e[(N+1)T] - e(0)$$

令 $N \rightarrow \infty$, 有

$$\begin{aligned} \sum_{n=0}^{\infty} \{e[(n+1)T] - e(nT)\} &= \lim_{N \rightarrow \infty} \{e[(N+1)T] - e(0)\} \\ &= \lim_{n \rightarrow \infty} e(nT) - e(0) \end{aligned}$$

所以

$$\lim_{n \rightarrow \infty} e(nT) = \lim_{z \rightarrow 1} (z-1)E(z)$$

Z 变换的终值定理形式亦可表示为

$$e(\infty) = \lim_{n \rightarrow \infty} e(nT) = \lim_{z \rightarrow 1} (1 - z^{-1})E(z)$$

在离散系统分析中, 常采用终值定理求取系统输出序列的终值误差, 或称为稳态误差。

【例 3.8】 设 Z 变换函数为

$$E(z) = \frac{0.792z^2}{(z-1)(z^2 - 0.416z + 0.208)}$$

确定 $e(nT)$ 的终值。

解：由终值定理式(3.30)，得

$$\begin{aligned} e_{ss}(\infty) &= \lim_{z \rightarrow 1} (z-1) \cdot \frac{0.792z^2}{(z-1)(z^2 - 0.416z + 0.208)} \\ &= \lim_{z \rightarrow 1} \frac{0.792z^2}{z^2 - 0.416z + 0.208} = 1 \end{aligned}$$

(5) 卷积定理

设 $x(nT)$ 和 $y(nT)$ 为两个采样函数，其离散卷积定义为

$$x(nT) * y(nT) = \sum_{k=0}^{\infty} x(kT)y[(n-k)T] \quad (3.31)$$

若
则

$$\begin{aligned} g(nT) &= x(nT) * y(nT) \\ G(z) &= X(z) \cdot Y(z) \end{aligned} \quad (3.32)$$

证明：由 Z 变换，得

$$X(z) = \sum_{k=0}^{\infty} x(kT)z^{-k}, \quad Y(z) = \sum_{n=0}^{\infty} y(nT)z^{-n}$$

所以

$$X(z) \cdot Y(z) = \sum_{k=0}^{\infty} x(kT)z^{-k} Y(z)$$

根据平移定理及 Z 变换定义，有

$$z^{-k}Y(z) = Z\{y[(n-k)T]\} = \sum_{n=0}^{\infty} y[(n-k)T]z^{-n}$$

故

$$X(z) \cdot Y(z) = \sum_{k=0}^{\infty} x(kT) \sum_{n=0}^{\infty} y[(n-k)T]z^{-n}$$

交换求和次序并代入式(3.31)，上式可写为

$$\begin{aligned} X(z) \cdot Y(z) &= \sum_{n=0}^{\infty} \left\{ \sum_{k=0}^{\infty} x(kT)y[(n-k)T] \right\} z^{-n} \\ &= \sum_{n=0}^{\infty} [x(nT) * y(nT)] z^{-n} \end{aligned}$$

再由定理已知条件

$$G(z) = Z[g(nT)] = Z[x(nT) * y(nT)]$$

则

$$G(z) = X(z) \cdot Y(z)$$

卷积定理指出，两个采样函数卷积的 Z 变换，就等于该两个采样函数相应 Z 变换的乘积。在离散系统分析中，卷积是沟通时域与 z 域的桥梁。

3.3.3 Z 逆变换

由 $E(z)$ 求出采样函数 $e^*(t)$ 称为 Z 逆变换，除了直接查表外，常用的方法有长除法、部分分式展开法和留数算法。

在连续系统中，应用拉氏变换把描述系统的微分方程转换为 s 的代数方程，然后写出系统的传递函数，即可用拉氏逆变换法求出系统的时间响应，从而简化了系统的研究。与此类

似，在离散系统中应用 Z 变换，也是为了把 s 的超越方程或者描述离散系统的差分方程转换为 z 的代数方程，然后写出离散系统的脉冲传递函数 (z 传递函数)，再用 Z 逆变换法求出离散系统的时间响应。

已知 Z 变换表达式 $E(z)$ ，求相应离散序列 $e(nT)$ 的过程记为

$$e(nT) = Z^{-1}[E(z)] \quad (3.33)$$

当进行 Z 逆变换时，信号序列仍是单边的，即当 $n < 0$ 时， $e(nT) = 0$ 。

1. 长除法

长除法又称为综合除法、幂级数法。 Z 变换函数 $E(z)$ 通常可以表示为按 z^{-1} 升幂排列的两个多项式之比：

$$E(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots + b_m z^{-m}}{1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_n z^{-n}}, \quad m \geq n \quad (3.34)$$

其中， a_i ($i=1, 2, \cdots, n$) 和 b_j ($j=0, 1, \cdots, m$) 均为常系数。通过对式(3.34)直接进行综合除法，得到按 z^{-1} 升幂排列的幂级数展开式

$$E(z) = c_0 + c_1 z^{-1} + c_2 z^{-2} + \cdots + c_n z^{-n} + \cdots = \sum_{n=0}^{\infty} c_n z^{-n} \quad (3.35)$$

如果所得到的无穷幂级数是收敛的，则按 Z 变换定义可知，式(3.35)中的系数 c_n ($n=0, 1, \cdots, \infty$) 就是采样脉冲序列 $e^*(t)$ 的脉冲强度 $e(nT)$ ，因此可以直接写出 $e^*(t)$ 的脉冲序列表达式

$$e^*(t) = \sum_{n=0}^{\infty} c_n \delta(t - nT) \quad (3.36)$$

要从一组 $e(nT)$ 值中求出通项表达式，一般比较困难。但在实际应用中，通常只需要求出数值序列的前几项，因此用幂级数法计算比较简便，这也是 Z 变换法的优点之一。

【例 3.9】 设 Z 变换函数

$$E(z) = \frac{z^3 + 2z^2 + 1}{z^3 - 1.5z^2 + 0.5z}$$

试用长除法求 $E(z)$ 的 Z 逆变换。

解：将给定的 $E(z)$ 表示为

$$E(z) = \frac{1 + 2z^{-1} + z^{-3}}{1 - 1.5z^{-1} + 0.5z^{-2}}$$

利用综合除法，得

$$E(z) = 1 + 3.5z^{-1} + 4.75z^{-2} + 6.375z^{-3} + \cdots$$

由式(3.36)得采样函数

$$e^*(t) = \delta(t) + 3.5\delta(t - T) + 4.75\delta(t - 2T) + 6.375\delta(t - 3T) + \cdots$$

只要表示 Z 变换的无穷幂级数 $E(z)$ 在 z 平面的某个区域内是收敛的，则在应用 Z 变换法解决离散系统问题时，就不需要指出 $E(z)$ 在什么 z 值上收敛。

2. 部分分式展开法

使用部分分式展开法也可以求出采样函数 $x^*(t)$ 的数值序列表达式，求取 Z 逆变换的过程与求取拉氏逆变换的过程十分相似：先求出 $X(z)$ 的特征方程式的根，使用待定系数法把 $X(z)$ 的解析式转换成部分分式，最后求出各个分式所对应的采样函数之和。

设

$$X(z) = \frac{b_0 z^m + b_1 z^{m-1} + \cdots + b_m}{a_0 \prod_{i=1}^n (z - p_i)}$$

展开成

$$\frac{X(z)}{z} = \sum_{i=1}^n \frac{A_i}{z - p_i} \quad (3.37)$$

式中

$$A_i = [(z - p_i) \frac{X(z)}{z}]_{z=p_i}$$

则 Z 逆变换

$$x(kT) = Z^{-1} \left[\sum_{i=1}^n \frac{z A_i}{z - p_i} \right] \quad (3.38)$$

逐项查 Z 变换表, 最后写出已知 $E(z)$ 对应的采样函数

$$e^*(t) = \sum_{n=0}^{\infty} \sum_{i=1}^n e_i(nT) \delta(t - nT) \quad (3.39)$$

【例 3.10】 设 Z 变换函数为

$$E(z) = \frac{(1 - e^{-aT})z}{(z-1)(z - e^{-aT})}$$

试求其 Z 逆变换。

解: 因为

$$\frac{E(z)}{z} = \frac{1 - e^{-aT}}{(z-1)(z - e^{-aT})} = \frac{1}{z-1} - \frac{1}{z - e^{-aT}}$$

所以

$$E(z) = \frac{z}{z-1} - \frac{z}{z - e^{-aT}}$$

查 Z 变换表 (见表 3.2) 中的第 3 项及第 8 项知, 在采样瞬时相应的信号序列为

$$e(nT) = 1 - e^{-anT}$$

由式(3.39)得

$$e^*(t) = \sum_{n=0}^{\infty} (1 - e^{-anT}) \delta(t - nT)$$

相应地有

$$e(0) = 1$$

$$e(T) = 1 - e^{-aT}$$

$$e(2T) = 1 - e^{-2aT}$$

3. 留数算法

留数算法也称为反演积分法, 不仅可用于有理分式, 还可用于无法应用部分分式法及幂级数法来求 Z 逆变换的超越函数。

函数 $E(z)$ 可以看做复数 z 平面上的洛朗 (Laurent) 级数, 利用积分关系可求出级数的各项系数:

$$Z^{-1}[E(z)] = e(kT) = \frac{1}{2\pi j} \oint_C E(z) z^{k-1} dz \quad (3.40)$$

积分路径 C 应包括被积式中的所有极点。根据留数定理

$$e(kT) = \sum_{i=1}^n \text{Res}[E(z) z^{k-1}]_{z=p_i} \quad (3.41)$$

式中, n 表示极点数, p_i 表示第 i 个极点。

因为

$$\operatorname{Res}[E(z)z^{k-1}]_{z \rightarrow p_i} = \lim_{z \rightarrow p_i} (z - p_i)E(z)z^{k-1}$$

所以

$$e(kT) = \sum_{i=1}^n \lim_{z \rightarrow p_i} [(z - p_i)E(z)z^{k-1}]$$

【例 3.11】设 Z 变换函数

$$E(z) = \frac{z^2}{(z-1)(z-0.5)}$$

试用留数法求其 Z 逆变换。

解：因为函数

$$E(z)z^{n-1} = \frac{z^{n+1}}{(z-1)(z-0.5)}$$

有 $z_1 = 1$ 和 $z_2 = 0.5$ 两个极点，极点处留数

$$\begin{aligned}\operatorname{Res}\left[\frac{z^{n+1}}{(z-1)(z-0.5)}\right]_{z \rightarrow 1} &= \lim_{z \rightarrow 1} \left[\frac{(z-1)z^{n+1}}{(z-1)(z-0.5)}\right] = 2 \\ \operatorname{Res}\left[\frac{z^{n+1}}{(z-1)(z-0.5)}\right]_{z \rightarrow 0.5} &= \lim_{z \rightarrow 0.5} \left[\frac{(z-0.5)z^{n+1}}{(z-1)(z-0.5)}\right] = -(0.5)^n\end{aligned}$$

所以，由式(3.41)得

$$e(nT) = 2 - (0.5)^n$$

相应的采样函数

$$\begin{aligned}e^*(t) &= \sum_{n=0}^{\infty} e(nT)\delta(t-nT) = \sum_{n=0}^{\infty} [2 - (0.5)^n]\delta(t-nT) \\ &= \delta(t) + 1.5\delta(t-T) + 1.75\delta(t-2T) + 1.875\delta(t-3T) + \cdots\end{aligned}$$

顺便指出，关于函数 $E(z)z^{n-1}$ 在极点处的留数计算方法如下：

若 z_i ($i=1, 2, \cdots, k$) 为单极点，则

$$\operatorname{Res}[E(z)z^{n-1}]_{z \rightarrow z_i} = \lim_{z \rightarrow z_i} [(z - z_i)E(z)z^{n-1}] \quad (3.42)$$

若 $E(z)z^{n-1}$ 有 n 阶重极点 z_i ，则

$$\operatorname{Res}[E(z)z^{n-1}]_{z \rightarrow z_i} = \frac{1}{(n-1)!} \lim_{z \rightarrow z_i} \frac{d^{n-1}[(z - z_i)^n E(z)z^{n-1}]}{dz^{n-1}} \quad (3.43)$$

3.3.4 应用 Z 变换求解差分方程

用 Z 变换求解差分方程的步骤如下：

- ① 对差分方程两端进行 Z 变换；
- ② 利用 Z 变换的实数位移定理，将初始条件代入 Z 变换式，得到以 z 为变量的代数方程；
- ③ 将 Z 变换式写成有理多项式：

$$Y(z) = \frac{b_0 z^m + b_1 z^{m-1} + \cdots + b_m}{a_0 z^n + a_1 z^{n-1} + \cdots + a_n}$$

- ④ 对 $Y(z)$ 进行 Z 逆变换，得到差分方程的解。

【例 3.12】试用 Z 变换法解下列二阶差分方程：

$$c^*(t+2T) + 3c^*(t+T) + 2c^*(t) = 0$$

T 为单位时间时，化为

$$c(k+2) + 3c(k+1) + 2c(k) = 0$$

设初始条件 $c(0) = 0$, $c(1) = 1$ 。

解：对差分方程的每一项进行 Z 变换，根据实数位移定理：

$$Z[c(k+2)] = z^2 C(z) - z^2 c(0) - z c(1) = z^2 C(z) - z$$

$$Z[3c(k+1)] = 3z C(z) - 3z c(0) = 3z C(z)$$

$$Z[2c(k)] = 2C(z)$$

于是，差分方程变换为如下 z 代数方程

$$(z^2 + 3z + 2)C(z) = z$$

解出

$$C(z) = \frac{z}{z^2 + 3z + 2} = \frac{z}{z+1} - \frac{z}{z+2}$$

查 Z 变换表（见表 3.2），求出 Z 逆变换

$$c^*(t) = \sum_{n=0}^{\infty} [(-1)^n - (-2)^n] \delta(t - nT)$$

或写成

$$c(k) = (-1)^k - (-2)^k \quad k = 0, 1, 2, \dots$$

差分方程的解，可以提供线性定常离散系统在给定输入序列作用下的输出序列响应特性，但不便于研究系统参数变化对离散系统性能的影响。因此，还需要研究线性定常离散系统的另一种数学模型——脉冲传递函数。

3.4 脉冲传递函数

3.4.1 脉冲传递函数定义

与连续系统相似，线性离散系统脉冲传递函数的定义为零初始条件下系统或环节的输出采样函数的 Z 变换与输入采样函数 Z 变换之比。

设 $x(n)$, $y(n)$, $h(n)$ 分别表示系统的输入、输出和单位采样响应， $X(z)$, $Y(z)$, $H(z)$ 分别为它们的 Z 变换。

根据 Z 变换的定义和式(3.14)，有

$$\begin{aligned} Y(z) &= Z[y(n)] = \sum_{n=0}^{\infty} y(n) z^{-n} = \sum_{n=0}^{\infty} \left[\sum_{m=0}^{\infty} h(n-m) \cdot x(m) \right] \cdot z^{-n} \quad (k = n - m) \\ &= \sum_{m=0}^{\infty} x(m) \left[\sum_{n=0}^{\infty} h(n-m) z^{-n} \right] = \sum_{m=0}^{\infty} x(m) \left[\sum_{k=-m}^{\infty} h(k) z^{-k-m} \right] \\ &= \sum_{m=0}^{\infty} x(m) z^{-m} \left[\sum_{k=0}^{\infty} h(k) z^{-k} \right] = X(z) \cdot H(z) \end{aligned}$$

所以

$$H(z) = \frac{Y(z)}{X(z)} \quad (3.44)$$

即线性离散系统脉冲传递函数等于系统的单位采样响应的 Z 变换。

若系统可用线性差分方程描述：

$$\sum_{k=0}^n a_k y(n-k) = \sum_{r=0}^m b_r x(m-r)$$

取 Z 变换, 用滞后定理, 得

$$\sum_{k=0}^n a_k z^{-k} Y(z) = \sum_{r=0}^m b_r z^{-r} X(z)$$

则脉冲传递函数可表示为两个 z^{-1} 的多项式之比:

$$H(z) = \frac{\sum_{r=0}^m b_r z^{-r}}{\sum_{k=0}^n a_k z^{-k}} \quad (3.45)$$

差分方程、 Z 变换式和脉冲传递函数 $G(z)$, 都是对离散系统物理特性的不同数学描述, 虽然形式不同, 但实质不变, 并且可以根据以上关系相互转化。

【例 3.13】 设某环节的差分方程为

$$c(nT) = r[(n-k)T]$$

试求其脉冲传递函数 $G(z)$ 。

解: 对差分方程取 Z 变换, 并由实数位移定理, 得

$$C(z) = z^{-k} R(z)$$

$$G(z) = z^{-k}$$

当 $k=1$ 时, $G(z) = z^{-1}$, 在离散系统中其物理意义是代表一个延迟环节。它把其输入序列右移一个采样周期后再输出。

【例 3.14】 设开环系统中的传递函数

$$G(s) = \frac{a}{s(s+a)}$$

试求相应的脉冲传递函数 $G(z)$ 。

解: 将 $G(s)$ 展开成部分分式

$$G(s) = \frac{1}{s} - \frac{1}{s+a}$$

查 Z 变换表, 得

$$G(z) = \frac{z}{z-1} - \frac{z}{z-e^{-aT}} = \frac{z(1-e^{-aT})}{(z-1)(z-e^{-aT})}$$

线性定常离散控制系统稳定的充要条件是, 闭环系统脉冲传递函数特征方程的所有根(即闭环传递函数的所有极点)全都分布在 z 平面的单位圆内, 单位圆是稳定边界。

3.4.2 零阶保持器及其脉冲传递函数

当微机控制系统的输入量为连续的模拟量时, 需要通过 A/D 转换成离散的数字量, 一般 A/D 转换中都有采样保持环节, 其特性为零阶保持器。

设采样周期为 T , 零阶保持器的表达式为

$$h_0(t) = 1(t) - 1(t-T) \quad (3.46)$$

其拉氏变换为

$$H_0(s) = L[1(t) - 1(t - T)] = \frac{1 - e^{-Ts}}{s} \quad (3.47)$$

微机控制的输出信号经 D/A 后变成阶梯状模拟信号，再作用于被控对象，D/A 转换环节也具有零阶保持器特性，可以将零阶保持器和被控对象一起看做微机控制系统的连续部分（如图 3.5 所示），传递函数为

$$G(s) = \frac{1 - e^{-Ts}}{s} G_c(s) \quad (3.48)$$

式中， $G_c(s)$ 是被控对象的传递函数， $\frac{1 - e^{-Ts}}{s}$ 是零阶保持器的传递函数。

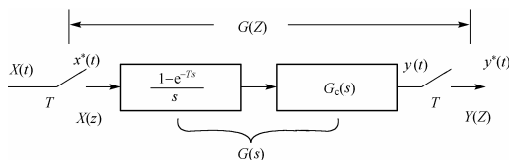


图 3.5 零阶保持器和被控对象的传递函数

在求系统的脉冲传递函数时，可先求出零阶保持器和被控对象连续部分的传递函数 $G(s)$ ，再求出它的脉冲响应函数 $h(t) = L^{-1}[G(s)]$ ， $h(t)$ 对应采样函数 $h^*(t)$ 的 Z 变换就是 $G(s)$ 的脉冲传递函数 $G(z)$ 。

【例 3.15】设离散系统如图 3.6(a) 所示，已知

$$G_c(s) = \frac{a}{s(s+a)}$$

试求系统的脉冲传递函数 $G(z)$ 。

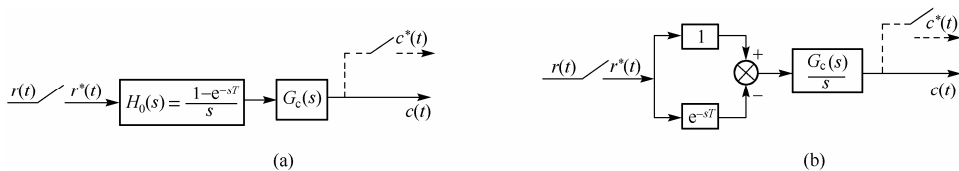


图 3.6 有零阶保持器的开环离散系统

解：将图 3.6(a) 等效转换为图 3.6(b)。

因为

$$\frac{G_c(s)}{s} = \frac{a}{s^2(s+a)} = \frac{1}{s^2} - \frac{1}{a} \left(\frac{1}{s} - \frac{1}{s+a} \right)$$

查 Z 变换表 3.2，有

$$\begin{aligned} Z \left[\frac{G_c(s)}{s} \right] &= \frac{Tz}{(z-1)^2} - \frac{1}{a} \left(\frac{z}{z-1} - \frac{z}{z-e^{-aT}} \right) \\ &= \frac{\frac{1}{a} z [(e^{-aT} + aT - 1)z + (1 - aTe^{-aT} - e^{-aT})]}{(z-1)^2(z-e^{-aT})} \end{aligned}$$

因此，有零阶保持器的开环系统脉冲传递函数

$$G(z) = (1 - z^{-1})Z \left[\frac{G_c(s)}{s} \right] = \frac{1}{a} \frac{[(e^{-aT} + aT - 1)z + (1 - aTe^{-aT} - e^{-aT})]}{(z - 1)(z - e^{-aT})}$$

由此可见，零阶保持器不影响离散系统脉冲传递函数的极点。

3.4.3 开环系统脉冲传递函数

如果开环离散系统由两个串联环节构成，则开环系统脉冲传递函数的求法与连续系统情况不完全相同。这是因为在两环节串联时，有两种不同的情况。

当两个串联环节之间有采样器隔开时，如图 3.7(a)所示，有

$$\begin{aligned} V(z) &= X(z) \cdot G_1(z) \\ Y(z) &= V(z) \cdot G_2(z) = X(z) \cdot G_1(z) \cdot G_2(z) \\ G(z) &= \frac{Y(z)}{X(z)} = G_1(z) \cdot G_2(z) \end{aligned} \quad (3.49)$$

当几个串联环节之间都用采样器隔开时，等效的脉冲传递函数等于几个环节的脉冲传递函数之积。

两个串联环节之间无采样器隔开，如图 3.7(b)所示。第二级的输入信号不是离散的，因此不能对第二级直接进行 Z 变换；须将第二级和第一级连在一起，连续部分的输入、输出皆为采样序列，才可以进行 Z 变换。因此

$$Y(z) = X(z) \cdot Z[G_1(s) \cdot G_2(s)] = X(z) \cdot G_1 G_2(z) \quad (3.50)$$

等效的脉冲传递函数为

$$G(z) = G_1 G_2(z) = Z[G_1(s) \cdot G_2(s)] \quad (3.51)$$

注意： Z 变换的乘积和传递函数乘积的 Z 变换是不同的。

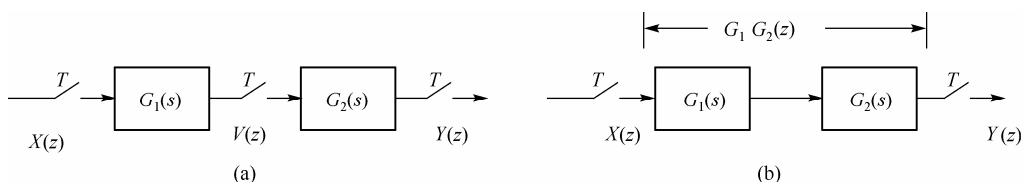


图 3.7 串联环节的两种形式

【例 3.16】 设开环离散系统如图 3.7(a), (b)所示，其中 $G_1(s) = 1/s$ ， $G_2(s) = a/(s + a)$ ，输入信号 $r(t) = 1(t)$ ，试求系统(a)和(b)的脉冲传递函数 $G(z)$ 和输出的 Z 变换 $C(z)$ 。

解：查 Z 变换表，输入 $r(t) = 1(t)$ 的 Z 变换为

$$R(z) = \frac{z}{z - 1}$$

对于系统(a)

$$G_1(z) = Z \left[\frac{1}{s} \right] = \frac{z}{z - 1}, \quad G_2(z) = Z \left[\frac{a}{s + a} \right] = \frac{az}{z - e^{-aT}}$$

因此

$$G(z) = G_1(z)G_2(z) = \frac{az^2}{(z - 1)(z - e^{-aT})}$$

$$C(z) = G(z)R(z) = \frac{az^3}{(z-1)^2(z-e^{-aT})}$$

对于系统(b)

$$G_1G_2(s) = \frac{a}{s(s+a)}$$

$$G(z) = G_1G_2(z) = Z\left[\frac{a}{s(s+a)}\right] = \frac{z(1-e^{-aT})}{(z-1)(z-e^{-aT})}$$

$$C(z) = G(z)R(z) = \frac{z^2(1-e^{-aT})}{(z-1)^2(z-e^{-aT})}$$

显然，在串联环节之间有无同步采样开关隔离时，其总的脉冲传递函数和输出 Z 变换是不同的。但是，不同之处仅表现在其零点不同，极点仍然一样。这也是离散系统特有的现象。

3.4.4 闭环系统脉冲传递函数

典型的微机闭环控制系统如图3.8所示，与连续系统的闭环传递函数相似，闭环脉冲传递函数是系统的输出信号与输入信号的 Z 变换之比。

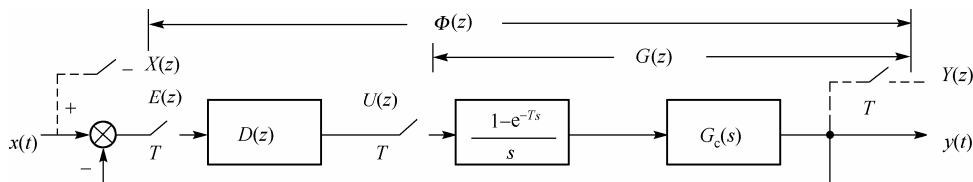


图 3.8 典型的微机闭环控制系统

输入 $x(t)$ 和输出 $y(t)$ 均为连续信号，设采样周期为 T ，它们对应的 Z 变换分别为 $X(z)$ 和 $Y(z)$ 。

$$Y(z) = U(z) \cdot G(z)$$

$$U(z) = E(z) \cdot D(z)$$

$$E(z) = X(z) - Y(z)$$

则

$$Y(z) = \frac{D(z) \cdot G(z)}{1 + D(z) \cdot G(z)} X(z)$$

$$\Phi_c(z) = \frac{E(z)}{X(z)} = \frac{1}{1 + D(z)G(z)}$$

即

$$\Phi(z) = \frac{Y(z)}{X(z)} = \frac{D(z) \cdot G(z)}{1 + D(z) \cdot G(z)} \quad (3.52)$$

式中， $D(z)$ 为控制器的脉冲传递函数， $\Phi_c(z)$ 为闭环离散系统对于输入量的误差脉冲传递函数， $G(z)$ 为广义对象（包括被控对象和零阶保持器）的脉冲传递函数， $\Phi_c(z)$ 和 $\Phi(z)$ 是研究离散系统常用的两个闭环脉冲传递函数。

需要指出的是，离散系统脉冲传递函数不能从 $\Phi(s)$ 和 $\Phi_c(s)$ 求 Z 变换得来，即

$$\Phi(z) \neq Z[\Phi(s)], \quad \Phi_c(z) \neq Z[\Phi_c(s)]$$

这种原因，也是由于采样器在闭环系统中有多种配置之故。

【例 3.17】设闭环离散系统结构图如图 3.9 所示，试证其闭环脉冲传递函数为

$$\Phi(z) = \frac{G_1(z)G_2(z)}{1 + G_1(z)HG_2(z)}$$

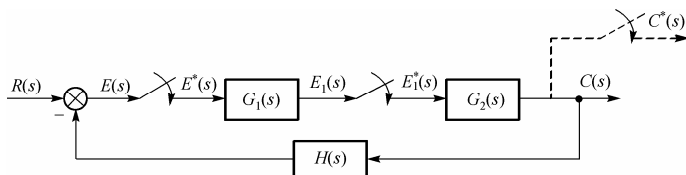


图 3.9 闭环离散系统

证明：由图 3.9 得

$$C(s) = G_2(s)E_1^*(s)$$

$$E_1(s) = G_1(s)E^*(s)$$

对 $E_1(s)$ 离散化，有

$$E_1^*(s) = G_1^*(s)E^*(s)$$

$$C(s) = G_2(s)G_1^*(s)E^*(s)$$

考虑到

$$E(s) = R(s) - H(s)C(s) = R(s) - H(s)G_2(s)G_1^*(s)E^*(s)$$

离散化后，有

$$E^*(s) = R^*(s) - HG_2^*(s)G_1^*(s)E^*(s)$$

即

$$E^*(s) = \frac{R^*(s)}{1 + G_1^*(s)HG_2^*(s)}$$

所以，输出信号的采样拉氏变换

$$C^*(s) = G_2^*(s)G_1^*(s)E^*(s) = \frac{G_1^*(s)G_2^*(s)R^*(s)}{1 + G_1^*(s)HG_2^*(s)}$$

对上式进行 Z 变换，证得

$$\Phi(z) = \frac{C(z)}{R(z)} = \frac{G_1(z)G_2(z)}{1 + G_1(z)HG_2(z)}$$

【例 3.18】设闭环离散系统结构图如图 3.10 所示，

试证其输出采样信号的 Z 变换函数

$$C(z) = \frac{RG(z)}{1 + GH(z)}$$

证明：由图 3.10 得

$$C(s) = G(s)E(s)$$

$$E(s) = R(s) - H(s)C^*(s)$$

所以

$$C(s) = G(s)R(s) - G(s)H(s)C^*(s)$$

对上式离散化，有

$$C^*(s) = GR^*(s) - HG^*(s)C^*(s)$$

解得

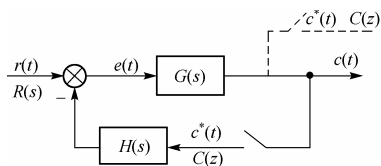


图 3.10 闭环离散系统

$$C^*(s) = \frac{GR^*(s)}{1 + HG^*(s)}$$

上式取 Z 变换，证得

$$C(z) = \frac{GR(z)}{1 + HG(z)}$$

由于从上式解不出 $C(z)/R(z)$ ，因此求不出脉冲传递函数，但可以求出闭环系统的采样输出信号 $c^*(t)$ 。

对于采样器在闭环系统中具有各种配置的闭环离散系统典型结构图，及其输出采样信号的 Z 变换函数 $C(z)$ ，可参见表 3.3。

表 3.3 典型闭环离散系统及输出 Z 变换传递函数

符号	系统结构图	$C(z)$ 计算式
1		$\frac{G(z)R(z)}{1 + GH(z)}$
2		$\frac{RG_1(z)G_2(z)}{1 + G_2HG_1(z)}$
3		$\frac{G(z)R(z)}{1 + G(z)H(z)}$
4		$\frac{G_1(z)G_2(z)R(z)}{1 + G_1(z)G_2H(z)}$
5		$\frac{RG_1(z)G_2(z)G_3(z)}{1 + G_2(z)G_1G_3H(z)}$
6		$\frac{RG(z)}{1 + HG(z)}$
7		$\frac{R(z)G(z)}{1 + G(z)H(z)}$
8		$\frac{G_1(z)G_2(z)R(z)}{1 + G_1(z)G_2(z)H(z)}$

习题与思考题

1 已知差分方程

$$y(kT) + y(kT - T) = x(kT) + 2x(kT - 2T)$$

输入序列为

$$x(kT) = \begin{cases} k & k \geq 0 \\ 0 & k < 0 \end{cases}$$

初始条件为

$$y(0) = 2$$

试用迭代法求解差分方程。

- 2 设序列为 $y(kT) = e^{-akT}$ ，求其 Z 变换 $y(z)$ 。

- 3 设序列为
$$x(kT) = \begin{cases} 1 & k \text{ 为奇数} \\ 0 & k \text{ 为偶数} \end{cases}$$

求其 Z 变换 $x(z)$ 。

- 4 求斜坡函数 $\text{tu}(t)$ 的 Z 变换。

- 5 分别用常除法、部分分式展开法和留数计算法求 $E(z)$ 的 Z 反变换：

$$E(z) = \frac{0.6z}{z^2 - 1.4z + 0.4}$$

- 6 设线性离散系统的差分方程为

$$y(kT) + 3y(kT - T) + 4y(kT - 2T) + 5y(kT - 3T) = x(kT) - 3x(kT - T) + 2x(kT - 2T)$$

且初始静止。试求系统的脉冲传递函数。

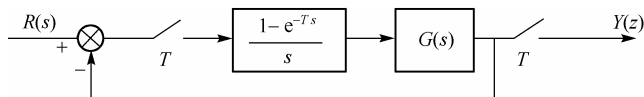
- 7 设线性离散系统的脉冲传递函数为

$$G(z) = \frac{z^4 + 3z^3 + 2z^2 + z + 1}{z^4 + 4z^3 + 5z^2 + 3z + 2}$$

且初始静止。试求系统的差分方程。

- 8 设系统框图分别如图 3.7(a)、(b)所示， $G_1(s) = 1/s$ ， $G_2(s) = 10/(s + 10)$ ，试求系统(a)、(b)的脉冲传递函数。

- 9 设系统框图如下图所示， $G(s) = K/s(s + a)(s + b)$ ， $T = 1\text{ s}$ 试求系统的闭环脉冲传递函数。



- 10 设微机闭环控制系统框图如图 3.8 所示，且 $D(z) = 1$ ， $T = 1$ ， $G_c(s) = \frac{1}{s(s + a)}$ ，输入为单位阶跃序列，试分析系统的过渡过程。

第 4 章 运动控制技术

在现代化的生活和生产等各个领域，都需要各种可控制的运动。大量的各种各样的电动机是最常用的运动源，而对电动机的控制则是实现运动控制最有效的方式。

对电动机的控制已经不仅仅是通过继电器和开关元件来实现的启动、制动、正反转等简单的控制，在不少场合还需要实现对转速、转角、转矩、电压、电流等物理量进行精确的复杂控制。

微机控制技术、微电子技术、电力电子技术、传感器技术和材料技术的飞速发展带来了运动控制技术的重大进展。

① 电路简单，灵活性和适应性强。采用微机控制使绝大多数控制逻辑可通过软件实现，不再需要复杂的逻辑电路和许多开关元件。如果需要修改控制规律，一般不必改变系统的硬件电路，只须修改程序即可。在系统调试和升级时，可以不断尝试选择最优参数，非常方便。

② 可以实现较复杂的控制。微处理器有更强的逻辑功能，运算速度快，精度高，有大容量的存储单元，因此有能力实现复杂的控制，如优化控制等。

③ 无零点漂移，控制精度高。不会出现模拟电路中经常遇到的零点漂移问题。不管被控量是大是小，都可以保证足够的控制精度。

④ 可提供灵活方便的人机界面，多机联网工作。

⑤ 可以采用多种嵌入式控制方式，例如，工业控制计算机、PLC、数字信号处理器（DSP）和各种微控制器（MCU）。

4.1 顺序控制

4.1.1 顺序控制原理

顺序控制是按规定的时间顺序或事件顺序进行动作的自动控制。

例如，现代化生产要求生产机械或生产过程在现场输入信号（按钮、行程开关、光电开关、温度信号、压力信号、时间信号）的作用下，按规定的转换条件实现有顺序的动作。

【例 4.1】冷加工自动线中钻孔动力头钻孔过程的顺序控制如图 4.1 所示。钻孔过程分为以下 5 步：

- ① 动力头在起始位置（行程开关 X_0 受压），按启动开关按钮 A 后，电磁阀 DT_1 通电，动力头快进；
 - ② 压上行程开关 X_1 ，使电磁阀 DT_2 通电（ DT_1 保持通电），动力头由快进转工进（钻孔）；
 - ③ 压上行程开关 X_2 ，使 DT_1 、 DT_2 断电，开始定时延迟，动力头原地镗削（精镗）；
 - ④ 延迟时间到， DT_3 通电，动力头快退；
 - ⑤ 动力头退回到原位，行程开关 X_0 又受压， DT_3 断电，动力头停止。
- 这样完成一个周期的循环动作之后，又返回到第 1 步，开始下一个循环的动作。

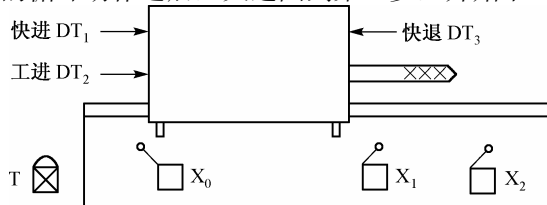


图 4.1 钻孔动力头控制示意图

在加工过程中，快进、钻孔、精镗、快退和停止等五个工作状态的顺序转换是由按钮、行程开关和时间信号等现场输入信号而决定的。

近年来，工业自动控制系统和机电一体化产品中的顺序控制系统，已经由继电器控制系统、数字逻辑电路组成的无触点专用逻辑控制系统，发展为以各种微型计算机为控制核心的专用和通用顺序控制系统（包括可编程顺序控制器）。

4.1.2 顺序控制系统特性

1. 开关量信号多

一般情况下，顺序控制系统的输入输出信号多数是二状态的开关量信号（如行程开关的闭合或断开，电磁阀的通电或断电等），而且数量较多。

2. 逻辑关系复杂

有些复杂的顺序控制系统输出信号和输入信号之间，各个输出信号之间存在复杂的逻辑关系和时序关系。例如，高层建筑的多台电梯群控系统，在接到乘客呼唤信号后，需要权衡尽快满足各楼层乘客的需要和尽可能节约能耗等多方面要求，进行最佳控制。

3. 定时范围大

机械动作相对来说比较缓慢，各个动作之间的间隔和持续时间从几秒至几十分钟，因此控制机械动作的定时范围比较大。

4. 可靠性要求高

相当多的顺序控制系统工作在工厂等现场，环境条件比较差，有的被控对象本身就是一个电磁干扰源（如大功率电磁阀），又要求系统能长时间不间断地运行，否则会影响生产效率，甚至引发事故。

4.1.3 专用顺序控制器设计

专用的顺序控制器应用于批量生产的特定自动装置中，应在满足使用要求前提下，尽可能地简化硬件结构，通过软件充分利用微机内部资源，加强功能并提高系统的可靠性，保证其性能价格比高于通用的顺序控制器。

在设计专用顺序控制器时，首先应根据被控对象、加工的工艺要求和机械动作的特性，将工作过程划分为各个工步，确定每一工步的定时控制或条件控制参数。定时控制参数包括各个动作的同步关系、先后次序和每一个动作的持续时间。条件控制参数通常是指各个动作和现场输入信号（如运动部件的位置、液体或气体的容量或压力、加热部件的温度等）的逻辑关系。另外，还必须解决现场信号的输入方法和被控对象的驱动方法等问题。

顺序控制器的基本功能是通过定时和对现场状态的逻辑判断，实时地控制执行机构的动作，其中两个主要软件模块是时序控制程序和逻辑条件判断程序。

1. 时序控制程序设计

顺序控制系统中可以有多个执行机构，在同一时刻允许一个或几个执行机构动作。可以根据工艺要求，画出各个执行机构控制通道的输出波形图，划分出各个不同的状态，把各状态的输出控制字和持续时间以表格形式存放于程序存储器之中。利用定时器产生作为时间基准的定时中断，在定时器的中断服务程序中对当前状态的持续时间进行计时。时间到即转入下一个状态，查表取出持续时间常数开始计时。通过输出控制字实现对操作的控制。

2. 逻辑条件判断程序设计

在顺序控制系统中，现场状态输入信号对某一个执行机构动作的控制有以下几种情况：

- ① 输入信号逻辑或控制：当某些输入信号中至少有一个有效时执行机构就动作；
- ② 输入信号逻辑与控制：当某些输入信号全部有效时执行机构才动作；
- ③ 输入信号的逻辑组合控制：一些输入信号中至少有一个有效而且另一些输入信号全部有效时执行机构才动作。

设计逻辑条件判断程序时，应为每一个输出通路设置逻辑条件控制字，分别控制每个执行机构的每个动作，例如，逻辑条件控制字的每一位对应于输入信号中的一个，其中为“1”的位参加逻辑操作：符合逻辑与条件必须是为“1”的位对应输入信号全部有效，符合逻辑或条件则是这些为“1”的位指出的输入信号中至少有一个有效。只有当逻辑与、逻辑或均符合条件时，对应输出通路才进行逻辑条件符合处理，使执行机构动作。

4.1.4 通用顺序控制器的选用

如果自行设计顺序控制系统不一定能保证较好的性能价格比、较短的开发时间和合理的开发成本，则应针对应用特点，提出合理的技术指标，如输入/输出的通路数量、工步的定时范围，输出信号和输入信号的逻辑关系、动作的精确程度、系统可靠性等，选用适当的通用顺序控制器，如 PLC。

4.2 程 序 控 制

4.2.1 程序控制原理

程序控制是根据输入的指令和数据，按规定的顺序、运动轨迹、运动距离和运动速度等规律而完成工作的自动控制。

程序控制是为了满足高精度、高效率加工复杂零件的需要而发展起来的。按运动轨迹的控制要求，可分为点位控制和轮廓（直线、曲线与曲面）控制。

点位控制的主要控制目标是准确达到本次运动的终点坐标，对运动的路径、速度等都没有特殊规定，所以它的控制电路功能只需要记忆应走的移动量和已走的移动量，并将所记忆的两个移动量进行比较，当两个移动量比较差值为零时，即认为到达本次运动的终点。

轮廓控制要求运动过程的中间点形成某种指定形状，因此必须用有限的机械运动方向来实现多种多样的运动轨迹。大型精密加工机床采用光电编码器（码盘）、光栅、感应同步器作为反馈测量元件构成闭环数字程序控制来达到要求，但其结构复杂，难以调整和维护。较为简单的方法是利用开环数字程序控制进行各种插补运算来解决这一矛盾。开环数字程序控制框图如图 4.2 所示。

插补器实际上是一个函数发生器，能按给定的基本数据产生一定的函数曲线，并以增量形式（如脉冲）向各坐标连续输出，实现给定的运动轨迹。用插补器完成的插补计算就是按给定的基本数据（如直线或圆弧的坐标等），插补（插值）中间坐标数据，从而把曲线形状描述出来的一种计算。

按插补器的功能可以分成平面直线插补器、圆弧插补器、非圆二次曲线插补器，以及空间直线和圆弧插补器等。利用直线和圆弧两种插补器可得到大部分运动轨迹，因此应用也最多。

最常用的插补计算是逐点比较插补法，也可采用数字积分

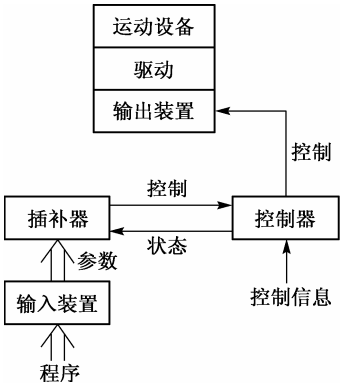


图 4.2 开环数字程序控制框图

器插补法（又称为数字微分分析法，DDA 法）、数字脉冲乘法器插补法（又称为 MIT 法）等。

逐点比较法是以阶梯折线来逼近直线或圆弧等曲线的。执行机构每走一步，都要和给定轨迹上的坐标值进行一次比较，看当前位置是在给定轨迹的上方或下方，或是在给定轨迹的里面或外面，从而确定下一步的进给方向。如果在给定轨迹的下方，则下一步向给定轨迹的上方走，反之向下方走；如果在给定轨迹的里面，则下一步向给定轨迹的外面走，反之向里面走。如此走一步后再算一步，决定下一步走向，从而使折线逼近给定的轨迹，即形成逐点比较插补。

逐点比较插补法产生的阶梯折线与规定的加工直线或圆弧之间的最大误差为一个脉冲当量（如步进电机步距），只要把脉冲当量取得足够小，就可达到要求的精度。

4.2.2 逐点比较法直线插补

1. 直线插补计算原理

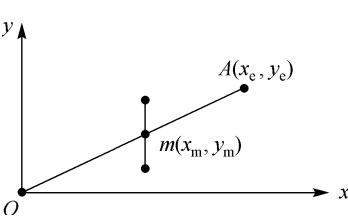
(1) 插补计算过程

直线插补计算时，每走一步，都要进行以下四次算术逻辑运算：

- ① 偏差判别：即判别偏差 $F \geq 0$ 或 $F < 0$ ，根据该逻辑运算的结果，确定下一步进给方向和新的偏差计算；
- ② 坐标进给：根据所在象限及偏差符号，决定沿哪个坐标轴，是沿正向还是负向进给；
- ③ 偏差计算：进给一步后，计算新的加工点对给定轨迹的偏差，作为下一次偏差判别的依据；
- ④ 终点判别：每进给一步，终点计数器减 1，判断是否到达终点。若未到达终点，则进行下一步计算。如此不断循环，直至到达终点为止。

(2) 偏差计算

假定运动轨迹为直线 OA （如图 4.3 所示）。取直线起点为坐标原点，终点坐标 (x_e, y_e) 是已知的。 $m(x_m, y_m)$ 为加工点（动点）。若 m 在 OA 直线上，则根据相似三角形关系可得



$$\frac{x_m}{y_m} = \frac{x_e}{y_e}$$

(4.1)

即

$$y_m x_e - x_m y_e = 0$$

定义直线插补的偏差判别式为

$$F_m = y_m x_e - x_m y_e$$

(4.2)

图 4.3 运动轨迹为直线的偏差计算

若 $F_m = 0$ ，则 m 点就在 OA 直线上；
若 $F_m > 0$ ，则 m 点在 OA 直线上方；

若 $F_m < 0$ ，则 m 点在 OA 直线下方。

由此可得逐点比较法的第一象限直线插补原理：从直线的起点（即坐标原点）出发，当 $F_m \geq 0$ 时，沿 $+x$ 方向走一步；当 $F_m < 0$ 时，沿 $+y$ 轴方向走一步。当两方向所走的步数与终点坐标 (x_e, y_e) 相等时，发出终点到信号，停止插补。

利用递推方法，可以减少计算偏差的乘法次数。

对于第一象限而言，设加工点正处于 m 点。

若 $F_m \geq 0$ ，表明 m 点在 OA 线上或在 OA 上方，应沿 $+x$ 方向进给一步，走一步以后新的坐标值为

$$x_{m+1} = x_m + 1$$

$$y_{m+1} = y_m$$

新点的偏差为 $F_{m+1} = F_m - y_e$ 。

若 $F_m < 0$ ，表明 m 点在 OA 的下方，应向+y 方向进给一步，走一步后新的坐标为

$$x_{m+1} = x_m$$

$$y_{m+1} = y_m + 1$$

新点的偏差为 $F_{m+1} = F_m + x_e$ 。

这样，运动的起点是坐标原点，起点的偏差为 $F_0 = 0$ ，动点的偏差都可以由前一点偏差 F_m 和终点坐标相加或相减得到，操作大为简单。

(3) 终点判断方法

逐点比较法的终点判断方法有以下几种：

- ① 设置 x, y 两个减法计数器，在运动开始前， x, y 计数器初值为终点坐标 x_e, y_e ，在 x 坐标（或 y 坐标）进给一步时 x 计数器（或 y 计数器）减 1，当两个计数器都减到 0 时，到达终点；
- ② 设置一个终点计数器，初值为 x_e, y_e 坐标给定的总步数，每当 x 或 y 坐标方向进给一步时，终点计数器减 1，当计数器减为零时，到达终点。

2. 不同象限的直线插补

四个象限直线插补的偏差计算公式和进给方向如表 4.1 所示。表中计算公式中的 x_e, y_e 均取四个象限直线终点坐标值的绝对值。

表 4.1 直线插补的偏差计算公式和进给方向

$F_m \geq 0$			$F_m < 0$		
所在象限	进给方向	偏差计算	所在象限	进给方向	偏差计算
一、四	+x	$F_{m+1} = F_m - y_e$	一、二	+y	$F_{m+1} = F_m + x_e$
二、三	-x		三、四	-y	

3. 直线插补计算的程序实现

(1) 数据的输入及存放

在计算机的内存中开辟 6 个单元 XE, YE, NXY, FM, XOY 和 ZF，分别存放终点横坐标 x_e 、终点纵坐标 y_e 、总步数 N_{xy} 、动点偏差 F_m 、直线所在象限值 xoy 和走步方向标志 Z_f 。

其中， $N_{xy} = N_x + N_y$ ， xoy 等于 1, 2, 3, 4 分别代表第一象限、第二象限、第三象限和第四象限， xoy 的值可由终点坐标(x_e, y_e)的正负符号来确定， F_m 的初值为 $F_0 = 0$ ， $Z_f = 1, 2, 3, 4$ 分别代表走步方向为 +x, -x, +y, -y。

(2) 直线插补计算的程序流程

实现直线插补计算 4 个步骤的程序流程如图 4.4 所示。

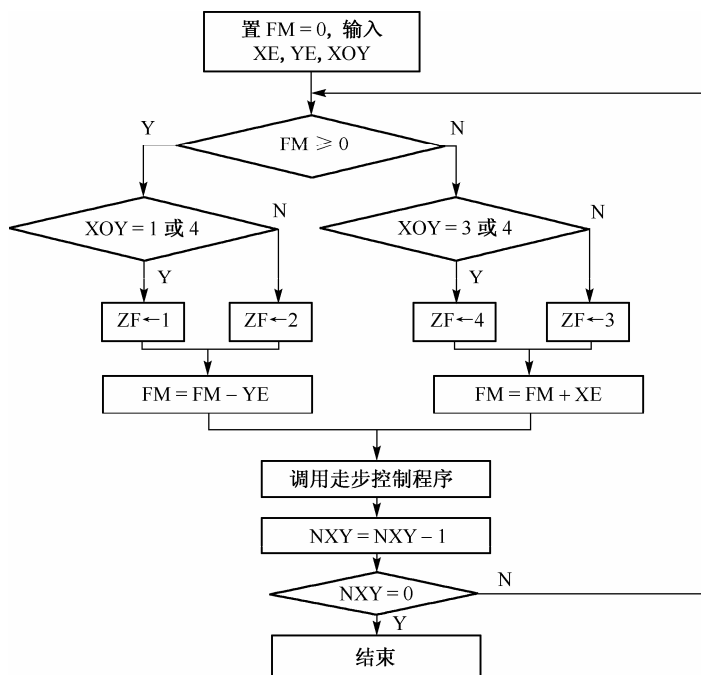


图 4.4 直线插补计算程序流程图

【例 4.2】设加工第一象限直线 OA ，起点为 $O(0, 0)$ ，终点坐标为 $A(6, 4)$ ，进行插补计算并绘出走步轨迹图。

解：坐标进给的总步数 $N_{xy} = |6 - 0| + |4 - 0| = 10$ ， $x_e = 6$ ， $y_e = 4$ ， $F_0 = 0$ ， $xoy = 1$ 。插补计算过程如表 4.2 所示，走步轨迹图如图 4.5 所示。

表 4.2 插补计算过程

步 数	偏差判别	坐标进给	偏差计算	终点判断
起点			$F_0 = 0$	$N_{xy} = 10$
1	$F_0 = 0$	$+x$	$F_1 = F_0 - y_e = -4$	$N_{xy} = 9$
2	$F_1 < 0$	$+y$	$F_2 = F_1 + x_e = 2$	$N_{xy} = 8$
3	$F_2 > 0$	$+x$	$F_3 = F_2 - y_e = -2$	$N_{xy} = 7$
4	$F_3 < 0$	$+y$	$F_4 = F_3 + x_e = 4$	$N_{xy} = 6$
5	$F_4 > 0$	$+x$	$F_5 = F_4 - y_e = 0$	$N_{xy} = 5$
6	$F_5 = 0$	$+x$	$F_6 = F_5 - y_e = -4$	$N_{xy} = 4$
7	$F_6 < 0$	$+y$	$F_7 = F_6 + x_e = 2$	$N_{xy} = 3$
8	$F_7 > 0$	$+x$	$F_8 = F_7 - y_e = -2$	$N_{xy} = 2$
9	$F_8 < 0$	$+y$	$F_9 = F_8 + x_e = 4$	$N_{xy} = 1$
10	$F_9 > 0$	$+x$	$F_{10} = F_9 - y_e = 0$	$N_{xy} = 0$

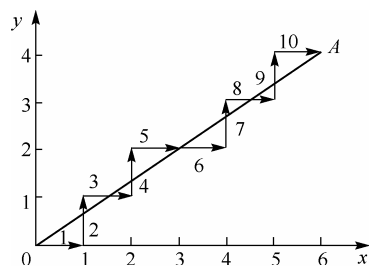


图 4.5 直线插补走步轨迹图

4.2.3 逐点比较法圆弧插补

1. 圆弧插补计算原理

(1) 偏差计算

设运动轨迹为第一象限内的逆时针圆弧 \overline{AB} ，圆弧的圆心在坐标原点，并已知圆弧的起

点为 $A(x_0, y_0)$ ，终点 $B(x_e, y_e)$ ，圆弧半径为 R 。动点 $m(x_m, y_m)$ 与圆心的距离为 R_m ，比较 R_m 和 R （实际上是比较它们的平方值），就可以判别偏差。

$$R_m^2 = x_m^2 + y_m^2$$

$$R^2 = x_0^2 + y_0^2$$

偏差判别式可定义为

$$F_m = R_m^2 - R^2 = x_m^2 + y_m^2 - R^2 \quad (4.3)$$

若 $F_m = 0$ ，则 m 点在圆弧上；若 $F_m > 0$ ，则 m 点在圆弧外；若 $F_m < 0$ ，则 m 点在圆弧内。

由此可得逐点比较法第一象限圆弧插补的原理：从圆弧的起点出发，当 $F_m \geq 0$ 时，沿 $-x$ 方向走一步，当 $F_m < 0$ 时，沿 $+y$ 轴方向走一步。当两个方向所走的步数与终点坐标 (x_e, y_e) 相等时，发出终点到信号，停止插补。

同样，可以利用递推方法来简化计算。

若 $F_m \geq 0$ ，表明 m 点在圆弧上或在圆弧外，应沿 $-x$ 方向进给一步至 $m+1$ 点，新的坐标值为

$$x_{m+1} = x_m - 1$$

$$y_{m+1} = y_m$$

新点的偏差为 $F_{m+1} = F_m - 2x_m + 1$ 。

若 $F_m < 0$ ，表明 m 点在圆弧内，应向 $+y$ 方向进给一步至 $m+1$ 点，新的坐标值为

$$x_{m+1} = x_m$$

$$y_{m+1} = y_m + 1$$

新点的偏差为 $F_{m+1} = F_m + 2y_m + 1$ 。

这样，运动的起点是圆弧的起点，偏差为 $F_0 = 0$ 。动点的偏差都可以由前一点的坐标和偏差 F_m 相加或相减得到，同样使操作大为简单。

(2) 终点判断方法

与直线插补的终点判断相似，将 x 方向的走步数 $|x_e - x_0|$ 和 y 方向的走步数 $|y_e - y_0|$ 之和，即总步数 N_{xy} 作为终点计数器的初值，每当 x 或 y 坐标方向进给一步时，终点计数器减 1，当计数器减为零时，到达终点。

(3) 插补计算过程

圆弧插补计算过程分为 5 个步骤：偏差判别、坐标进给、偏差计算、坐标计算、终点判断。比直线插补计算过程多了一步，即需要计算动点坐标值。

2. 不同象限的圆弧插补

四个象限圆弧插补的偏差计算公式和进给方向如表 4.3 所示。表中用 SR_1, SR_2, SR_3, SR_4 和 NR_1, NR_2, NR_3, NR_4 分别表示第一至第四象限的顺圆弧和逆圆弧。

3. 圆弧插补计算的程序实现

(1) 数据的输入及存放

在计算机的内存中开辟 8 个单元 $XO, YO, NXY, FM, RNS, XM, YM$ 和 ZF ，分别存放起点的坐标 x_0, y_0 、总步数 N_{xy} 、加工点偏差 F_m 、圆弧种类值 RNS 、动点坐标 x_m, y_m 和走步方向标志 Z_f 。其中 RNS 等于 1, 2, 3, 4 和 5, 6, 7, 8 分别代表 SR_1, SR_2, SR_3, SR_4 和 NR_1, NR_2, NR_3, NR_4 ，

RNS 的值可由起点和终点的坐标的正负符号来确定, F_m 的初值为 $F_0=0$, x_m 和 y_m 的初值为 x_0 和 y_0 , 走步方向标志 $Z_f=1, 2, 3, 4$ 分别表示走步方向为 $+x, -x, +y, -y$ 。

表 4.3 圆弧插补的偏差计算公式和进给方向

偏 差	圆 弧 种 类	进 给 方 向	偏 差 计 算	坐 标 计 算
$F_m \geq 0$	SR_1, NR_2	$-y$	$F_{m+1} = F_m - 2y_m + 1$	$x_{m+1} = x_m$ $y_{m+1} = y_m - 1$
	SR_3, NR_4	$+y$		
	NR_1, SR_4	$-x$	$F_{m+1} = F_m - 2x_m + 1$	$x_{m+1} = x_m - 1$ $y_{m+1} = y_m$
	NR_3, SR_2	$+x$		
$F_m < 0$	SR_1, NR_4	$+x$	$F_{m+1} = F_m + 2x_m + 1$	$x_{m+1} = x_m + 1$ $y_{m+1} = y_m$
	SR_3, NR_2	$-x$		
	NR_1, SR_2	$+y$	$F_{m+1} = F_m + 2y_m + 1$	$x_{m+1} = x_m$ $y_{m+1} = y_m + 1$
	NR_3, SR_4	$-y$		

(2) 圆弧插补计算的程序流程
按照上述圆弧插补计算过程 5 个步骤实现的程序流程如图 4.6 所示。

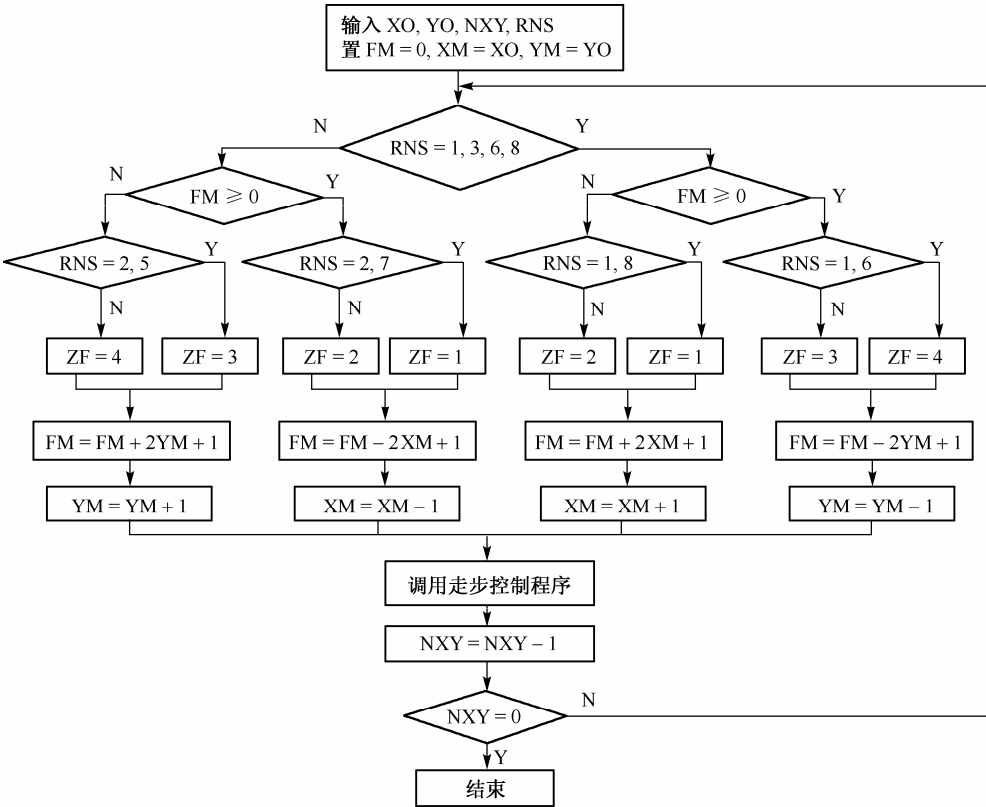


图 4.6 圆弧插补计算程序流程图

【例 4.3】 已知第一象限逆圆弧 \overline{AB} 起点的坐标为 $A(4, 0)$, 终点的坐标为 $B(0, 4)$, 进行插补计算, 绘出走步轨迹图。
解: 插补计算过程如表 4.4 所示, 走步轨迹如图 4.7 所示。

表 4.4 圆弧插补计算过程

步 数	偏差判 别	坐标进 给	偏 差 计 算	坐 标 计 算	终 点 判 断
起点			$F_0 = 0$	$x_0 = 4, y_0 = 0$	$N_{xy} = 8$
1	$F_0 = 0$	$-x$	$F_1 = F_0 - 2x_0 + 1 = -7$	$x_1 = x_0 - 1 = 3, y_1 = 0$	$N_{xy} = 7$
2	$F_1 < 0$	$+y$	$F_2 = F_1 + 2y_1 + 1 = -6$	$x_2 = 3, y_2 = y_1 + 1 = 1$	$N_{xy} = 6$
3	$F_2 < 0$	$+y$	$F_3 = F_2 + 2y_2 + 1 = -3$	$x_3 = 3, y_3 = y_2 + 1 = 2$	$N_{xy} = 5$
4	$F_3 < 0$	$+y$	$F_4 = F_3 + 2y_3 + 1 = 2$	$x_4 = 3, y_4 = y_3 + 1 = 3$	$N_{xy} = 4$
5	$F_4 > 0$	$-x$	$F_5 = F_4 - 2x_4 + 1 = -3$	$x_5 = x_4 - 1 = 2, y_5 = 3$	$N_{xy} = 3$
6	$F_5 > 0$	$+y$	$F_6 = F_5 + 2y_5 + 1 = 4$	$x_6 = 2, y_6 = y_5 + 1 = 4$	$N_{xy} = 2$
7	$F_6 > 0$	$-x$	$F_7 = F_6 - 2x_6 + 1 = 1$	$x_7 = x_6 - 1 = 1, y_7 = 4$	$N_{xy} = 1$
8	$F_7 > 0$	$-x$	$F_8 = F_7 - 2x_7 + 1 = 0$	$x_8 = x_7 - 1 = 0, y_8 = 4$	$N_{xy} = 0$

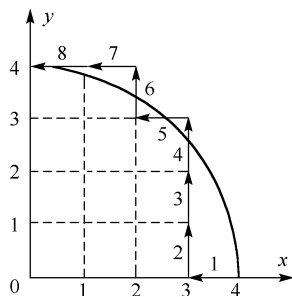


图 4.7 圆弧插补走步轨迹图

4.2.4 运动控制芯片

随着微型计算机控制技术的发展,近年来出现了 MCX314 系列、PCL6045 系列等多种运动控制芯片,可以方便地解决十分复杂的程序控制问题。例如,匀速和变速脉冲的发射、升降速规划、直线和圆弧插补、原点及限位开关管理、编码器计数、失步检测等,都只需主控器向运动控制芯片发出简单指令即可完成,大大简化了运动控制系统的软硬件结构和开发工作。由于运动控制芯片简单易用,性能优越,价格也富于竞争力,因而在数控机床、电脑雕刻机、电脑绣花机、工业机器人、医用设备、自动仓库、绕线机、绘图仪、点胶机、IC 电路制造设备、芯片装片机、IC 电路板制作等各种程序控制中得到了广泛的实际运用,并正在不断拓展新的运用领域。

1. MCX314 系列运动控制芯片的主要性能

MCX314 系列运动控制芯片能与 8 位或 16 位数据总线接口,通过命令、数据和状态等寄存器实现 4 轴 3 联动的位置、速度、加速度等的运动控制和实时监控,实现直线、圆弧、位元 3 种模式的轨迹插补,输出脉冲频率达 4 MHz。每轴都有伺服反馈输入端、4 个输入点和 8 个输出点,能独立地设置为恒速、线性或 S 曲线加/减速控制方式,并有两个 32 位的逻辑、实际位置计数器和状态比较寄存器,实现位置的闭环控制。

以低功耗 3.3 V MCX314AL 芯片为例,其主要特性如下:

(1) 4 轴独立控制,自带 32 位圆弧/直线插补。

- 直线插补
 - 范围 各轴 $-8\ 388\ 607 \sim +8\ 388\ 607$
 - 速度 $1 \sim 4\ \text{Mpps}$
 - 位置精密度 $\pm 0.5\text{LSB}$ 以下(在全插补范围内)
- 圆弧插补
 - 范围 各个轴 $-8\ 388\ 607 \sim +8\ 388\ 607$
 - 速度 $1 \sim 4\ \text{Mpps}$
 - 位置精密度 $\pm 1\text{LSB}$ 以下(在全插补范围内)
- 位模式插补
 - 速度 $1 \sim 4\ \text{Mpps}$ (依靠 CPU 数据设定时间)
- 其他功能
 - 线性常数,连续插补,插补步进传送(命令/外部信号),复数芯片多轴直线插补,非对称 S/T 型加减速

(2) 动作电源电压 $\pm 3.3\ \text{V}$ CPU,输出/入信号电平可以连接 CMOS, TTL,数据总线长度 8/16 位可选;当输入时钟 16 MHz (标准),最高输出速率为 4 Mpps;当 CLK=32 MHz 时,最高输出速率可达 8 Mpps;144 脚 LQFP 封装。

2. PCL6045 系列运动控制芯片的主要性能

PCL6045 系列运动控制芯片具有优秀的多轴插补控制功能，一个芯片就可以控制多个交流和步进/伺服电机。例如，PCL6045B 具有控制多达四个交流和步进/伺服电机所需的一切功能。当执行圆弧运动时，所有插补计算由芯片完成，上位机只需写入圆弧的参数即可。有预置运动位置的 FIFO 缓冲器，存储了三段运动数据，完成当前运动任务后，芯片自动将下段运动的数据移至当前目标位置寄存器，并在过渡脉冲外保持各轴速度的连续性。每完成一段运动，PCL6045B 会设置标志位并主动通知上位机，上位机只需在下两段运动完成前填入数据即可，并具有自动补偿间隙及在伺服到位前减少振荡的功能，从而使上位机的查询工作大大减轻，可以同时处理其他工作并保证各轴运动的连续性。

以 PCL6045B 芯片为例，其主要特性如下：

① 可独立控制 4 轴交流或步进/伺服电机，2, 3, 4 轴直线插补。

- 多段直线和圆弧间连续插补运动；任意两轴圆弧插补由芯片内部执行；
- 梯形和 S 形曲线运动，频率最高 6.5 MHz；
- 具有独特的 FIFO（先入先出）缓冲功能，内含三个寄存器组，可使芯片内存储三段运动数据；预置寄存器不但可存三点位置，而且可存每轴速度、加速度等数值；
- 13 种回零方式可满足各种要求，可对多达四个编码器信号进行解码并记录位置偏差，硬件比较器可在设定位置处产生触发和中断，伺服控制时到位前减少振荡，自动间隙补偿功能可改善运动精度。

② 动作电源电压： $\pm 5\text{ V} \pm 5\%$ （最大电流 10 mA）、 $\pm 3.3\text{ V} \pm 5\%$ （最大电流 10 mA）；输入时钟 19.6608 MHz（标准）；176 针 QFP 封装。

4.3 可编程控制器

可编程控制器（PLC）是为了满足开关量逻辑控制的需要而发展起来的，最初被称为可编程逻辑控制器（Programmable Logic Controller, PLC）。1980 年，美国电气制造协会（NEMA）将其命名为可编程控制器（Programmable Controller, PC）。

国际电工委员会（IEC）作了如下定义：“可编程控制器是一种专为在工业环境下应用而设计的数字运算操作的电子系统，它采用可程序的存储器，在其内部执行逻辑运算、顺序控制、定时、计数和算术运算等操作指令，并通过数字式或模拟式的输入/输出控制各种类型的机械或生产过程。可编程控制器及其有关外部设备的设计原则是，易于与工业控制系统联成一个整体，易于扩充功能。”

以微型计算机为基础的 PLC 功能已经远远超出了定义的范围，从小规模的单机顺序控制装置发展成为应用广泛、型号繁多的工业控制设备，不仅用于开关量控制，还可具有模拟量等各种类型信息的输入输出、计算、控制、调节、容错、联网等功能。由于其可靠性高，硬件连接简单，已经成为各种环境中最常用的现场控制装置之一，在各类控制网络中直接控制、驱动现场设备的底层节点。

4.3.1 PLC 的主要功能与特点

1. 主要功能

① 实现开关量的逻辑控制、顺序控制，用与（AND）、或（OR）、非（NOT）等逻辑指令，取代传统的继电器控制系统。

② 可以提供几十个甚至几千个计时器的定时控制, 计时值在编制程序时设定, 也可在现场实时设定, 实际计时值也可以实时地读出和修改。

③ 可以提供几十个甚至几千个计数器的计数控制, 高速计数频率可达几百 kHz 以上。

④ 可以具有模/数 (A/D) 转换和数/模 (D/A) 转换功能, 能完成过程控制所需要的对模拟量的检测、控制和调节, 例如, 对温度、压力、流量等连续变化的模拟量的闭环 PID 控制。

⑤ 可以提供拖动步进电机或伺服电机的单轴或多轴控制模块, 可广泛地应用于各种机械的位置控制。

⑥ 可以提供若干个移位寄存器, 用于各道工序的步序控制。

⑦ 具有数学运算 (矩阵运算、函数运算、逻辑运算等)、数据传递、数据转换、排序和查表、位操作等数据处理功能, 还能完成数据采集、分析、处理, 这些数据可通过通信接口传送到其他智能装置。

⑧ 可以实现 PLC 相互之间、PLC 与上位计算机、PLC 与其他智能设备间的信息交换。可以直接或通过通信处理单元、通信转接器与通用计算机相连构成网络。

⑨ 具有对系统异常情况进行识别、记忆或在发生异常情况时自动终止运行的监控功能。

⑩ 具有显示、打印、报警、对数据和程序硬拷贝等其他功能。

2. 特点

PLC 之所以能成为当今增长速度最快的工业自动控制设备, 除了上述强大功能外, 还由于它具备了许多独特的优点, 较好地解决了工业控制领域普遍关心的可靠、安全、灵活、方便、经济等问题, 其主要特点如下:

① 擅长处理大量的开关量和数字量, 能随时捕捉多种输入信号的变化, 分别进行状态组合运算和实时处理, 适应生产要求。

② 可靠性高, 抗干扰能力强, 可以连续工作几十万小时无故障。输入部件均带有光电耦合电路等隔离措施及滤波、电平转换、信号锁存等电路, 可以排除干扰信号并适应于工业现场; 输出部件具有输出锁存、显示、电平转换功能和可以直接驱动执行元件的大功率器件, 如机械触点式的继电器, 无触点的交流开关 (如双向可控硅) 及直流开关 (如晶体三极管) 等; 输入输出均通过接线端子排与现场设备连接, 能方便地用于实际控制系统。

③ 编程简单易学。可以采用类似于继电器控制线路的梯形图, 也可以采用类似于微机汇编指令的助记符语言, 还可以采用顺序功能图 (Sequential Function Chart, SFC) 语言, 用状态转移图来描述复杂的工程步进控制工艺。

④ 使用方便灵活。用软件功能取代了大量硬件和接线, 设计、安装容易; 程序的大部分可以在实验室模拟调试, 大大缩短了调试周期; 维修的工作量小, 查找故障和更换模块都很简单。

⑤ 有系列化、标准化、通用化的产品和模块可供选用, 功能齐全, 体积小, 能耗低。特别适应嵌入式控制的要求。既有功能简单、分散、廉价的小型装置, 又有高速度、高可靠性、多功能、网络化的大型装置。

⑥ 与工控机相比, 其缺点是软件资源较贫乏, 在要求快速、实时性强、模型复杂性高的领域不占有优势; 各厂商产品不能兼容, 开发较为困难。

4.3.2 PLC的结构与工作原理

1. 硬件结构

PLC 硬件结构框图如图4.8所示。系统硬件结构采用总线模块化结构, 根据系统中各个组

成部分的功能不同，制成具有统一标准总线的独立模块。只要根据系统的功能要求，选用相应的模块组装，就可以配置构成完整的系统。CPU 模块为系统的核心，基本 I/O 模块（模拟或数字）提供现场输入设备和输出设备与 CPU 的接口电路，智能接口模块提供与其他智能控制设备的接口，网络与通信接口模块提供和其他 PLC 或计算机系统的通信接口。

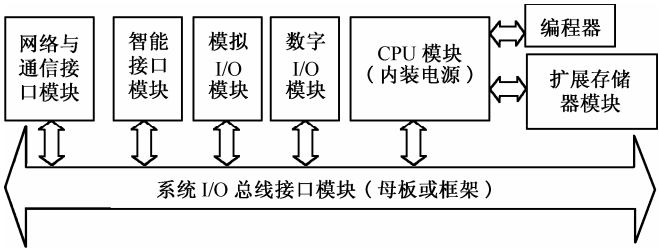


图 4.8 PLC 系统硬件结构框图

(1) CPU 模块

CPU 模块是 PLC 的核心部件，它包括电源、微处理器、系统存储器、控制逻辑和接口电路等，其主要功能是处理用户程序，接收来自输入模块的状态或数据，进行逻辑判断、数值计算、数据处理、输出状态控制命令，完成对整个系统的自诊断、内部工作状态监控和编程器管理等工作。

(2) 存储器模块

存储器包括系统存储器和用户存储器。系统存储器由生产厂家事先编写并固化好，其内容主要为监控程序、模块化应用功能子程序、命令解释和功能子程序的调用管理程序及各种参数等。用户存储器又分为程序区和数据区：程序区用于存放用户编写的控制程序；数据区用于存放输入、输出数据和中间变量，提供计时器、计数器、寄存器等，还存放满足系统程序使用和管理需要的系统状态和标志信息。

(3) 编程器

编程器可供现场编制、编辑、调试、监控用户程序，它通过通信接口与 CPU 联系，完成人机对话。有简易手持编程器、智能化图形编程器两种：简易型编程器采用命令语句助记符联机编程，智能型编程器常采用梯形图语言，并可脱机编程。编程和调试工作还可以利用编程和仿真软件在 PC 机上进行。

(4) I/O 模块

I/O 模块包括数字 I/O 模块和模拟 I/O 模块两类。

数字 I/O 模块是 PLC 与现场的开关量信号及电气控制设备之间的接口部件，可提供多种操作电平并具有一定的驱动能力，其输出类型可为直流的晶体管、交直流的继电器、交流的可控硅等方式。

模拟 I/O 模块可实现对连续参数的检测和控制，主要包括信号变换电路、模/数转换或数/模转换及隔离锁存电路。

(5) 智能接口模块

常用的智能接口模块包括远程 I/O 模块、高速计数模块、自动位置控制模块、PID 模块、中断控制模块和 ASCII 模块。

① 远程 I/O 模块，可以使 I/O 模块安装在远离 CPU 模块的用户设备旁，保证信号传送的可靠性，并节省大量电缆和安装费用。

② 高速计数模块，可以实现硬件计数，检测处理快速数字脉冲。

- ③ 自动位置控制模块，可实现高精度定位控制。
- ④ PID 模块，可提供多回路 PID 控制。
- ⑤ 中断控制模块，可提供优先级中断输入响应。
- ⑥ ASCII 模块，可连接具有 RS-232/RS-422 或 20 mA 电流环接口的各种外部设备，扩大 PLC 对数据的处理能力，等等。

智能模块其实就是独立的、具有基本功能和 PLC 总线接口的小型微处理器系统。这些模块在 PLC 的统一管理下可完成某些独立的特定功能，既可弥补 I/O 模块的不足，又不加重中央处理器的负担。

- (6) 网络通信接口模块
提供 PLC 之间、PLC 与计算机之间的通信接口，组成主从式通信网络或分布式计算机控制系统。
- (7) 系统 I/O 总线接口模块

系统 I/O 总线接口模块也称为母线或模块框架，是带有插槽的框架或母板，内部装有由总线接口电路、驱动电路等组成的印刷线路板，用以实现各插槽间的电气连接，将各种模块安装、组合成一个 PLC 系统。

2. 工作原理
多种输入信号状态的变化往往时刻不同，长短不一。为了及时捕捉、处理这些变化，并且对多种输入信号分别进行状态组合运算，PLC 采用了不断循环，顺序扫描的工作方式。每次扫描包括 4 个阶段：初始化、输入、执行和输出，如图 4.9 所示。

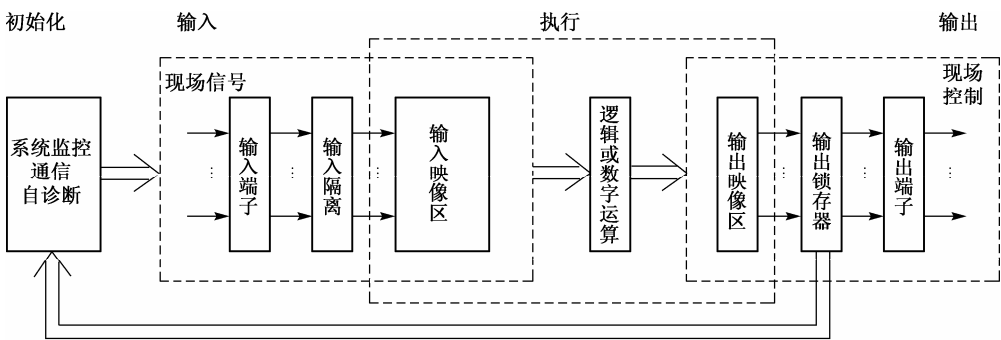


图 4.9 PLC 工作过程示意图

- (1) 初始化阶段
进行系统监控和自诊断，检查及复位监视定时器 WDT；与编程器交换信息。可以通过编程器修改内存程序、启动或停止 CPU、读 CPU 状态、封锁或开放输入/输出、对逻辑变量和数字变量进行读写。
如果配有数字处理器 DPU、网络，则与它们交换信息。
- (2) 采样输入阶段
以扫描的方式顺序读入所有输入端的信号状态，并存入输入映像区（又称为输入状态寄存器），其位数与输入端子数目相对应。
扫描读入的周期应小于各个输入信号电平保持时间的最小值，以保证能够及时捕捉到任何一个输入信号的状态变化，满足顺序控制要求随时检测每个信号状态变化的条件。

(3) 执行阶段

按照用户程序在存储器中的存放地址，从头至尾顺序扫描执行整个用户程序。

用户程序只能按照指令读取输入映像区的位的状态，而不能改变它的状态。在执行阶段里，即使输入信号的状态发生变化，输入映像区的内容也不会改变。只有到下一个工作周期的输入采样阶段，新的状态才被读入输入映像区。

若程序需要读入某输出状态，则从输出映像区的对应位读入，然后进行逻辑运算或数字运算。运算结果保存在输出映像区。

整个执行阶段只与输入输出映像区或其他内部数据区交换数据，并不直接访问 I/O 模块；输出映像区的内容，会随程序的执行而变化，但不直接影响输出端子的工作现状。只有当用户程序顺序执行完后，控制器才进入输出阶段。

(4) 输出阶段

将输出映像区中的所有输出状态同时转存到输出锁存器中，并驱动形成 PLC 的实际输出。

在一个扫描周期执行完毕后，地址计数器又恢复到初始地址，由初始化阶段开始下一个扫描周期。只要 PLC 的扫描周期小于各个输出执行机构的最小动作时间，就可以保证系统在整个运行期间，对任一输出点都可以做到连续控制，避免失控问题。

由图 4.9 可知：

- 输入映像区的内容，由输入阶段的输入端子状态决定；
- 输出映像区的状态，由程序执行阶段的程序执行结果决定；
- 输出锁存电路的状态，由程序执行结束后输出映像区的内容确定；
- 输出端子状态，由输出锁存电路确定。

这种程序运行与实际输入、输出分时进行的工作方式，使程序运行过程与现场外部设备在时间和空间都进行了有效的电气隔离，大大提高了可靠性和抗干扰能力，但也带来了输入/输出响应的滞后。

3. 响应时间

PLC 的输入端有一个输入信号发生变化到输出端对此做出反应所需的时间，称为响应时间。

响应时间主要与下列因素有关：

- 输入延迟，例如输入滤波时间常数；
- 输出延迟，例如输出继电器的机械滞后；
- 程序执行时间，受用户程序编程技巧等因素影响。

最短响应时间 = 输入延迟时间 + 一个扫描周期 + 输出延迟时间

最长响应时间 = 输入延迟时间 + (扫描周期 + 输出延迟时间) × 2

一般的工业控制是允许这些滞后的，对于实时响应要求较高的系统，可选用快速响应模块，执行高速处理指令，将其结果直接输往外部，从而不受循环扫描方式的制约。还可以利用高速脉冲计数模块，或利用中断控制功能，使某些信号得到迅速响应。

4.3.3 PLC常用编程语言

1. 梯形图语言

梯形图语言是在原电气控制系统中常用的接触器、继电器梯形图基础上演变而来的，它与电气操作原理相呼应，形象直观，为电气技术人员所熟知，是 PLC 的主要编程语言之一。

电气控制梯形图使用的是物理继电器、定时/计数器等硬接线，而 PLC 的梯形图的接点和线圈均为“软”继电器，实际上是系统存储器中的对应位。当该位为“1”时，表示线圈被激励、常开触点被闭合或常闭触点被断开；当该位为“0”时，表示的动作与上述相反。“软”的继电器、定时器/计数器等，均是通过软件实现控制的，因而使用方便，修改灵活，这是物理器件无法比拟的。

PLC 的每个梯形图网络由多个梯级组成，每个输出线圈可构成一个梯级。每个梯级可由多个支路组成，每个支路最右边的元素必须是输出线圈。PLC 梯形图从上至下按行绘制，两边的竖线类似电器控制图的电源线，称为母线（Busbar）。每一行左侧总是输入接点，并且把并联接点多的支路靠近最左端。输入接点在图形符号上只用常开和常闭来表示，而与其物理属性无关。输出线圈用圆形或椭圆形表示。

PLC 以扫描方式从左到右，从上到下的顺序执行用户的程序。扫描过程按梯形图的梯级顺序执行，上一个梯级的结果是下一个梯级的条件。

各种型号的 PLC 对一些特殊功能（如定时器，计数器等）的各种梯形图符号及编程方法各有其具体规定。

【例 4.4】实现如下控制过程：当开关量 1 和 2 均为“1”，且开关量 3 为“0”时，红灯亮；开关量 4 和 5 任一个为“1”时，绿灯亮。

解：采用继电器控制逻辑实现该控制的过程如下：
当常开触点 X1 和 X2 闭合且常闭触点 X3 闭合时，线圈 Y1 接通，Y1 触点闭合，红灯亮；当常开触点 X4 或 X5 中任何一个闭合时，线圈 Y2 接通，同时 Y2 触点接通，绿灯亮。其电路图如图 4.10 所示。

采用 PLC 实现该控制的过程如下：
先读入开关量 X1, X2, X3, X4, X5 的信息，然后对 X1, X2, X3 的状态进行逻辑运算，若逻辑条件满足，则输出开关量 Y1 为“1”，使外电路中红灯亮。同时对 X4、X5 状态进行逻辑运算，若条件满足，则输出开关量 Y2 为“1”，使外电路中绿灯亮。梯形图程序如图 4.11 所示。

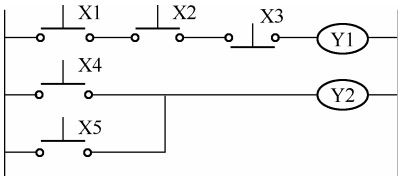


图 4.10 继电器控制电路图

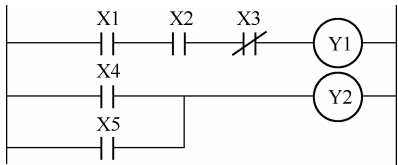


图 4.11 PLC 梯形图程序

从图 4.11 可以看出，梯形图中接点的串联相当于“逻辑与”（AND）：X1 和 X2 必须同时接通，同时 X3 断开，线圈 Y1 才能接通。接点的并联表示“逻辑或”（OR），即 X4 和 X5 中只要有一个接通，线圈 Y2 就能接通。

2. 助记符语言（命令语句表语言）
助记符语言类似于微机的汇编指令，一段梯形图也可以用一系列命令语句表来表示。命令语句由操作码和操作数组成。由于没有国际统一标准，所以各厂家所使用的命令语句对操作码的定义也不同。

对具有编制微机程序基础的人员来说，助记符语言键入比较方便，并且编程支路中元素的数量一般不受限制。通常用户程序从存储器的开始地址起连续不断地编程，并按地址号递增方向存放，中间不留空地址。

表4.5中分别用三菱公司 PLC 的命令语句和 GE 公司 PLC 的命令语句编写了可以实现图 4.10 功能的程序。

表 4.5 实现图 4.10 功能的程序

序 号	三菱公司 PLC 的命令语句		GE 公司 PLC 的命令语句		注 释
	操 作 码	操 作 数	操 作 码	操 作 数	
000	LD	X1	STR	X1	逻辑行开始，取输入 X1（常开接点）
001	AND	X2	AND	X2	串联结点 X2（常开接点）
002	ANI	X3	AND NOT	X3	串联结点 X3（常闭接点）
003	OUT	Y1	OUT	Y1	输出 Y1，本逻辑行结束
004	LD	X4	STR	X4	逻辑行开始，取输入 X4（常开接点）
005	OR	X5	OR	X5	并联结点 X5（常开接点）
006	OUT	Y2	OUT	Y2	输出 Y2，本逻辑行结束

3. 顺序功能图语言

状态转移图可以简单、直观地描述控制系统控制过程，在具有较复杂的工程步进控制工艺的场合下使用，可大幅度缩短设计时间。

顺序功能图语言（Sequential Function Chart, SFC）采用状态转移图方式编程。将系统的工作过程分为若干个阶段，这些阶段称为“状态”（State）。状态与状态之间由“转换”（Transition）分隔，相邻的状态具有不同的“动作”（Action）。当相邻的两状态之间的转换条件得到满足时，转换得以实现，即上一状态的动作结束而下一状态的动作开始，因而不会出现状态的动作重叠。当系统正处于某一状态时，将该状态称为“活动状态”。

状态用方框表示，方框内为状态元件号或状态名称；状态之间用线段连接，线段上的垂直短线和它旁边标注的文字符号或逻辑表达式表示状态的转移条件；该状态期间的输出信号用圆形、方框或椭圆形表示。

【例 4.5】小车运动的示意图如图 4.12 所示，画出实现该运动时的顺序功能图。

解：小车的工作过程如下：

- 当启动按钮后，小车前进；
- 当小车到右限位开关 X011 时，后退；
- 当小车到左限位开关 X013 时，停止，定时器 T0 开始计时；
- 定时器 T0 计时 5 s 后，小车第二次前进；
- 当小车运行到当右限位开关 X012 时第二次后退，运行到达左限位 X013 后，回到初态，进入下一个循环。

小车的前进及后退都是由同一电机的不同方向转动来带动的，因此加上联锁后，得到采用顺序功能图语言描述的状态转移图如图 4.13 所示。

- S0 为初始状态，当启动按钮后，状态从 S0 向 S20 转移，小车前进 Y021 动作；
- 当右限位开关 X011 接通时，状态 S20 向 S21 转移，前进输出 Y021 切断，后退输出 Y023 接通；
- 当左限位开关 X013 接通时，状态 S21 向 S22 转移，启动定时器 T0；
- 延时 5 s 后，定时器 T0 的触点动作，转至状态 S23，小车前进 Y021 动作；
- 当右限位开关 X012 接通时，状态 S23 向 S24 转移，前进输出 Y021 切断，后退输出 Y023 接通。到达左限位 X013 接通，状态返回 S0，又进入下一个循环。

3. 控制系统结构的选择

用 PLC 构成的控制系统有集中控制、远程控制和分布式控制 3 种结构。

集中控制一般采用本地 I/O 通道, 便于管理, 但缺点是现场的 I/O 电缆线要全部引入控制室, 故敷线和查线困难, 一般适用于控制规模较小的场合。

远程 I/O 控制, 可以克服敷线和查线的困难, 而且由于连接线短, 成本相应会降低, 但要增加远程驱动器的辅助电源, 因而适用于较大的系统。

分布式控制采用几台较小 PLC 分别控制某些设备, 再用通信线路将它们连接成一个网络控制系统, 需要附加通信硬件和软件, 因此用于控制要求较高或规模很大的场合。

4.4 直流电动机控制

从能量供给的角度来看, 微型计算机主要是从时间而不是从幅度对直流电动机进行控制的, 因而具有下列特点:

1. 功能增强

用于电动机控制的微机增加了多项功能, 例如, 脉宽调制(Pulse Width Modulation, PWM)、比较和捕捉、A/D 转换器、看门狗、各种串行总线接口等。

PWM 口经初始化设定后会自动地发出 PWM 控制信号, CPU 只在需要调整参数时才介入, 这已在直流电动机控制中广泛应用。

捕捉功能在电动机控制中可用于测频, 相当于在最初单片机中用计数器与外部中断联合测频功能。

集成了输入模拟信号必需的 A/D 转换器, 可用于直流测速发电机, 或测量电动机绕组的输出电压或电流等有模拟信号存在的情况。

电动机是一个电磁干扰源, 电动机的启停还会影响电网电压的波动, 它周围的电器开关也会引发火花干扰。因此, 除了采用必要的隔离、屏蔽和电路板合理布线等措施外, 看门狗的功能格外重要。看门狗在工作时不断地监视程序运行的情况, 一旦程序“跑飞”, 可以立即复位。

各种串行总线由于具有使用的信号线少(如 SPI 使用 5 条信号线, I²C 总线只使用两条信号线), 电路占用的面积大大减小, 信号受干扰的可能性小, 可利用的引脚相对增多等优点, 大有取代并行总线之势。在电动机的控制中, 使用有串行总线接口的芯片连接外部存储器及键盘和显示器等人机界面, 可大大减小电路的尺寸, 降低成本。

2. 速度更快

为了对电动机进行实时控制, 新一代微机采用了流水线技术, 执行指令与提取指令可同时进行完成。也有些采用了精简指令系统计算机(Reduced Instruction Set Computer, RISC)结构技术, 使指令执行的速度大大提高。

3. 小型化和低功耗

采用电动机控制专用的 DSP, 具有高速信号处理能力和数字控制功能, 还有为电动机控制应用所必需的外围功能, 因而可以实现如矢量控制、直接转矩控制、自适应控制等复杂算法。

4. 智能功率集成

将驱动电路、逻辑控制电路、检测和诊断电路、保护电路和半导体功率器件集成在一块芯片上的功率集成电路, 是电力电子技术与微电子技术相结合的产物, 它使功率器件具有智能功能, 因此又称为智能功率集成电路。

与分立功率器件组成的电动机驱动电路相比，智能功率集成电路有如下优点：

① 体积小，重量轻，但功能强大，甚至可以构成将驱动器安装在内部的电子电动机。

② 电器元件数量和连线减少，减小了分布电容和分布电感及信号传输的延时，从而提高了系统抗干扰的能力和可靠性。

③ 检测电路、诊断电路、保护电路、控制电路和功率电路内部集成在一起，使监控更易实现，大大降低了系统成本。

④ 集成化使电路的连线减少，减小了分布电容和分布电感及信号传输的延时，从而提高了系统抗干扰的能力；还可以使电路参数优化，避免在使用分立元件时，因不同厂商的产品所带来的兼容性问题。

半导体功率器件发展的另一个方向是智能功率模块（Intelligent Power Module, IPM）。它是将多个（或单个）功率器件组成半桥或全桥，并集成了快速恢复二极管、栅极（或基极）驱动电路和保护电路而形成的混合模块。

4.4.1 直流电动机调速原理

直流电动机是最早出现的电动机，由于它具有良好的线性调速特性，控制简单，效率高，动态特性优异，长期以来一直占据着调速控制的统治地位。

近年来，直流电动机的结构和控制方式都发生了很大的变化。随着计算机进入控制领域，以及新型的电力电子功率元器件的不断出现，采用全控型的开关功率元件进行 PWM 控制已成为主流方式。

随着永磁材料和工艺的发展，用永磁材料代替励磁部分的永磁直流电动机，体积小，结构简单，省电，已在中小功率范围内得到了广泛的应用。

1. 直流电动机调速原理

直流电动机转速 n 可由下式计算：

$$n = \frac{U - IR}{K\Phi} \quad (4.4)$$

式中 U ——电枢端电压；

I ——电枢电流；

R ——电枢电路总电阻；

Φ ——每极磁通量；

K ——电动机结构参数。

直流电动机的转速控制方法包括励磁磁通控制和电枢电压控制。

励磁磁通控制在低速时受磁极饱和的限制，在高速时受换向火花和换向器结构强度的限制，并且励磁线圈电感较大，动态响应较差。

大多数应用场合都采用在励磁恒定不变的情况下，通过调节电枢电压来实现调速。也可以采用基速以下调节电枢电压，基速以上削弱励磁磁通的联合控制法。

在对直流电动机电枢电压的控制和驱动中，对半导体功率器件的使用又可分为两种方式：线性放大驱动方式和开关驱动方式。

线性放大驱动方式使半导体功率器件工作在线性区，但此时会将大部分电功率损耗为热量。由于效率和散热问题严重，只能用于微小功率直流电动机的驱动。

绝大多数直流电动机采用开关驱动方式。开关驱动方式是使半导体功率器件工作在开关

状态，通过脉宽调制 PWM 来控制电动机电枢电压，实现调速。其原理图和输入输出电压波形如图 4.14 所示。

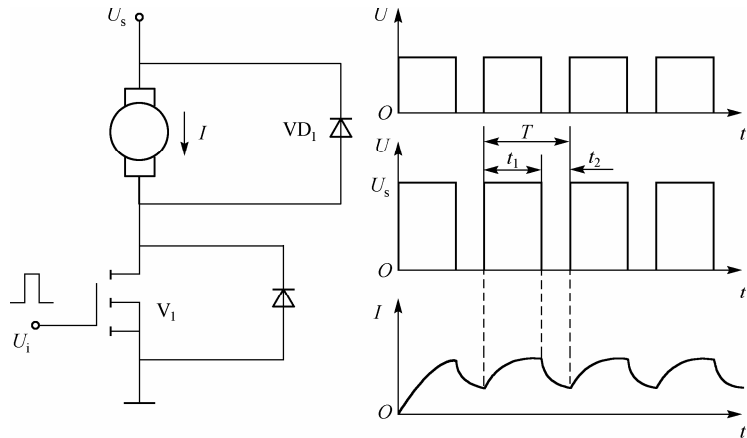


图 4.14 PWM 调速控制原理和电压电流波形图

图4.14中，当开关管 V_1 的栅极输入高电平时，开关管导通，直流电动机电枢绕组两端电压为 U_s 。 t_1 秒后，栅极输入变为低电平，开关管截止，电动机电枢两端电压为 0。 t_2 秒后，栅极输入重新变为高电平，开关管的动作重复前面的过程。直流电动机电枢绕组两端的电压波形随输入的电平高低而变化，其平均值 U_0 为

$$U_0 = \frac{t_1 U_s}{t_1 + t_2} = \frac{t_1}{T} U_s = \alpha U_s \tag{4.5}$$

式中，占空比 $\alpha = \frac{t_1}{T}$ 。

占空比 α 表示了一个周期 T 里，开关管导通的时间与周期的比值，其变化范围为 $0 \leq \alpha \leq 1$ 。在电源电压 U_s 不变的情况下，改变 α 值就可以改变端电压的平均值，从而达到调速的目的。

有如下三种方法可以改变占空比的值：

- (1) 定宽调频法
保持 t_1 不变，只改变 t_2 ，这样使周期 T （或频率）也随之改变。
- (2) 调宽调频法
保持 t_2 不变，而改变 t_1 ，这样使周期 T （或频率）也随之改变。
- (3) 定频调宽法
保持周期 T （或频率）不变，而同时改变 t_1 和 t_2 。

前两种方法的主要缺点是调速过程中当控制脉冲的频率与系统固有频率接近时，会引起振荡，因此用得很少。目前，主要采用定频调宽法控制直流电动机转速。

通过专用 PWM 集成电路或单片机的 PWM 口可以产生 PWM 控制信号，而分立的逻辑电子元件组成 PWM 信号电路或软件模拟法已经被淘汰。

根据直流电动机的转矩（电流）与转速的关系，电动机运行状态的四个象限如图 4.15 所示：第一象限是电动机正转运行状态，第三象限是电动机反转状态，第二象限和第四象限分别是电动机正转和反转时再生制动运行状态。控制方式和电路结构决定了电动机能在几个象限工作，功能较强的控制系统可以使电动机在四个象限都能运行。

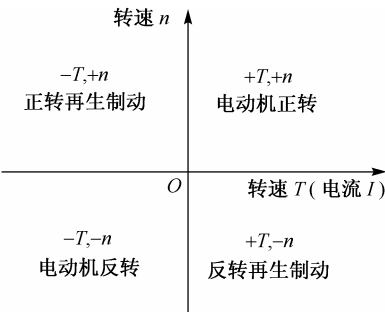


图 4.15 电动机四个运行象限

4.4.2 直流电动机不可逆PWM系统

直流电动机 PWM 控制系统有可逆和不可逆系统之分。可逆系统中电动机可以正反两个方向旋转，不可逆系统中电动机只能单向旋转。

1. 无制动的不可逆 PWM 系统

无制动不可逆 PWM 系统的特点是结构非常简单。由于在这种结构中电枢电流不能反向流动，也不能工作在制动状态，电动机只能进行单象限工作。

无制动不可逆 PWM 系统中，电枢电流的波形在每个 PWM 周期中由两段指数曲线组成： t_1 区间 V_1 导通，电枢绕组与电源接通，电流按指数规律上升，同时电枢绕组电感蓄能； t_2 区间 V_1 截止，电源断开，电枢绕组电感通过续流二极管 VD_1 释放能量，绕组中继续有电流按指数规律下降（如图 4.14 所示）。

PWM 控制方式中，直流电动机电枢电压波形为脉冲，电流波形为连续的波动。电流的波动将导致电动机输出转矩的波动。提高 PWM 频率可以大大地减小电流波动，从而减小转矩的波动。

图 4.16 所示的是一个使用单片机控制的不可逆 PWM 直流电动机闭环调速系统，它通过单片机的 PWM 口输出信号来控制直流电动机的转速。与电动机同轴安装的直流测速发电机，将转速信号通过单片机内部的 A/D 转换器进行 A/D 转换，然后将测速信号与给定转速进行比较，以决定占空比从而控制加减速。

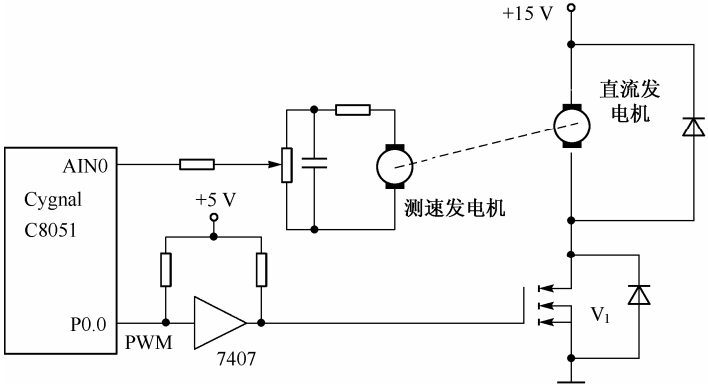


图 4.16 单片机控制的不可逆 PWM 系统

2. 有制动的不可逆 PWM 系统

为了产生制动作用，系统必须能在两个象限工作，为此增加一个开关管 V_2 ，给制动时的反向电流提供通路，如图 4.17 所示。

开关管 V_1 、 V_2 的 PWM 信号电平方向相反。在每个 PWM 周期的 t_1 区间， V_1 导通， V_2 截止，电流的路线和方向如图中虚线 1 所示，电动机工作在电动状态；在 t_2 区间， V_1 截止，电源被切断，电枢绕组的自感电动势使电流经过续流二极管 VD_2 形成回路，如虚线 2 所示；此时虽然开关管 V_2 的控制信号为高电平，由于续流二极管 VD_2 的钳位作用，使开关管 V_2 截止，其电流波形如图 4.18(a)所示。

在制动时，由于控制信号的 PWM 占空比不断减小，使电枢电压平均值 U_0 小于电动机的反电动势，电枢中的电流反向流动，产生制动转矩。在每个 PWM 周期的 t_1 区间，电枢绕组的自感电动势与反电动势之和大于电源电压，电流经过续流二极管 VD_1 将能量回馈给电源，电流的路线和方向如虚线 4 所示，电动机工作在再生发电制动状态；在 t_2 区间， V_2 在控制信号作用下导通，电流经过 V_2 形成回路，电流的路线和方向如虚线 3 所示，电动机处于能耗制动状态，制动时的电流波形如图 4.18(b) 所示。

当电动机轻载或空载时，电枢绕组中的电流很小。这时会出现电动和制动两种状态的交替：在每个 PWM 周期的 t_1 区间，电流先是按虚线 4 反向流动，电动机工作在再生发电制动状态；电流经过零点后，电源电压开始大于反电动势，电流按虚线 1 正向流动，电动机工作在电动状态。在 t_2 区间，由于 V_1 截止，电流先是按虚线 2 流动，电动机工作在续流电动状态；当续流降到零后，续流二极管 VD_2 失去钳位作用， V_2 导通，反电动势使电流改变方向，沿虚线 3 流动，电动机工作在能耗制动状态。轻载或空载时的电枢电流波形如图 4.18(c) 所示。

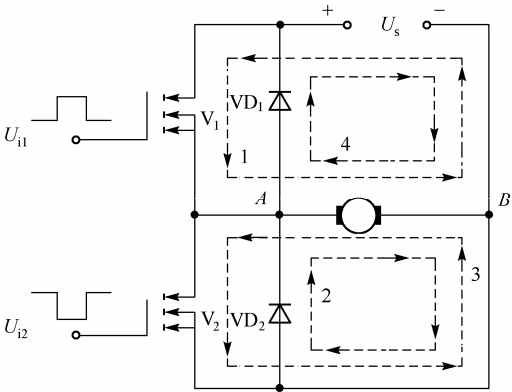


图 4.17 有制动的不可逆 PWM 驱动系统

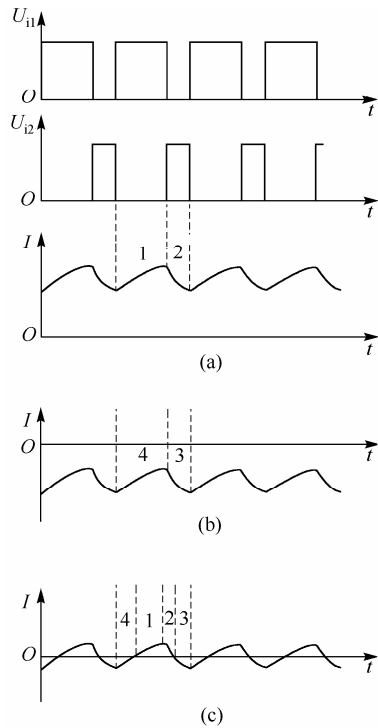


图 4.18 有制动的不可逆 PWM 系统电流波形

4.4.3 直流电动机可逆 PWM 系统

要求直流电动机正反转工作时，需要使用可逆 PWM 系统。可逆 PWM 系统分为双极性驱动和单极性驱动。

1. 双极性驱动可逆 PWM 系统

双极性驱动是指在一个 PWM 周期里，电动机电枢的电压极性呈正负变化。双极性驱动电路有 T 型电路和 H 型电路两种。

T 型电路由两个开关管组成，采用正负电源，相当于两个不可逆系统的组合。由于开关管要承受较高的反向电压，通常只用于低压小功率直流电动机驱动。

应用较多的 H 型电路如图 4.19 所示，4 个开关管分为 V_1V_4 组和 V_2V_3 组。同组的开关管同步导通或关断，不同组的开关管的导通与关断正好相反。

在每个 PWM 周期里，当控制信号为 U_{i1} 高电平、 U_{i2} 低电平时，开关管 V_1, V_4 导通， V_2, V_3 截止，电枢绕组承受从 A 到 B 的正向电压；当控制信号为 U_{i1} 低电平、 U_{i2} 高电平时，开关管 V_1, V_4 截止， V_2, V_3 导通，电枢绕组承受从 B 到 A 的反向电压。在一个 PWM 周期里电枢电压经历了正反两次变化（双极）。

电枢绕组所受的平均电压 U_0 取决于占空比 α ，由下式决定：

$$U_0 = \left(\frac{t_1}{T} - \frac{T-t_1}{T} \right) U_s = \left(\frac{2t_1}{T} - 1 \right) U_s = (2\alpha - 1) U_s \quad (4.6)$$

当 $0 \leq \alpha < 1/2$ 时, 电动机反转, 若 $\alpha = 0$, 则 $U_0 = -U_s$, 转速最大。

当 $1/2 < \alpha \leq 1$ 时, 电动机正转, 若 $\alpha = 1$, 则 $U_0 = U_s$, 转速最大。

当 $\alpha = 1/2$ 时, $U_0 = 0$, 电动机不转, 但电枢绕组中仍然有交变电流流动, 使电动机产生高频振荡, 这种振荡有利于克服电动机负载的静摩擦, 提高动态性能。

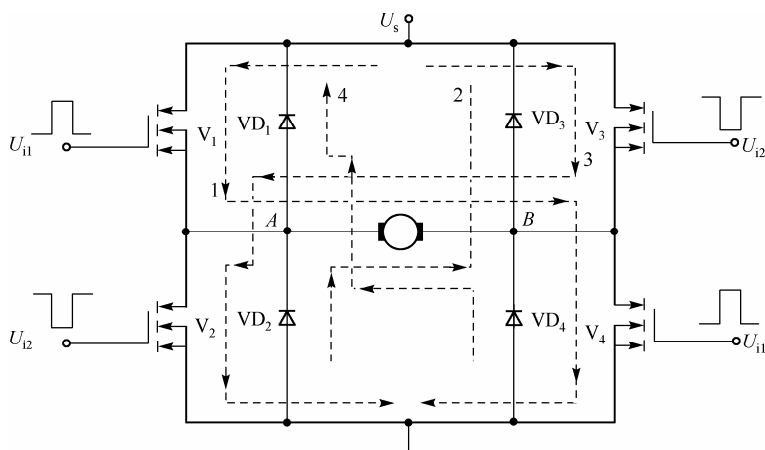


图 4.19 H 型双极可逆 PWM 驱动系统

电枢绕组中的电流波形如图 4.20 所示, 分以下 3 种情况:

① 电动机在较大负载情况下正转工作时的电流波形如图 4.20(a)所示。在每个 PWM 周期的 t_1 区间, V_1, V_4 导通, V_2, V_3 截止, 电枢绕组中电流的方向是从 A 到 B , 如图 4.19 中的虚线 1。 t_2 区间, V_2, V_3 导通, V_1, V_4 截止, 虽然电枢绕组加反向电压, 但由于绕组的负载电流较大, 电流的方向仍然不变, 但电流幅值的下降速率要比不可逆系统大, 因此电流的波动较大。

② 电动机在较大负载情况下反转工作时的电流波形如图 4.20(b)所示, 情形与正转时相反。

③ 当电动机在轻载下工作时, 工作状态呈电动和制动交替变化, 电枢电流很小, 波形基本上围绕横轴上下波动, 方向也不断地变化, 如图 4.20(c)所示。

区间 4: V_2, V_3 截止, 由于自感电动势的作用, 电枢中的电流维持原流向即从 B 到 A , 如图 4.19 中虚线 4, 经二极管 VD_1, VD_4 到电源, 电动机处于再生制动状态。由于二极管 VD_1, VD_4 的钳位作用, 此时 V_1, V_4 不能导通。

区间 1: 当电流衰减到零后, 二极管 VD_1, VD_4 失去钳位作用, 在电源电压的作用下, V_1, V_4 开始导通, 电流经 V_1, V_4 形成回路, 如图 4.19 中虚线 1 所示, 方向从 A 到 B , 处于电动状态。

区间 2: V_1, V_4 关断, 电枢电流在自感电动势的作用下经续流二极管 VD_2, VD_3 继续从 A 到 B , 流向如图 4.19 中虚线 2, 电动机仍处于电动状态。

区间 3: 当电流衰减为零后, V_2, V_3 开始导通, 电流回路如图 4.19 中的虚线 3, 电动机处于反接制动状态。

双极性驱动时, 电动机可在四个象限上工作。低速时的高频振荡有利于消除负载的静摩擦, 低速平稳性好, 但在工作的过程中电流波动大。由于 4 个开关管都处在开关状态, 功率损耗较大。由于必须通过增加死区来避免开关管直通的危险, 因此限制了开关频率的提高。

所以，双极性驱动只用于中小功率直流电动机的控制。

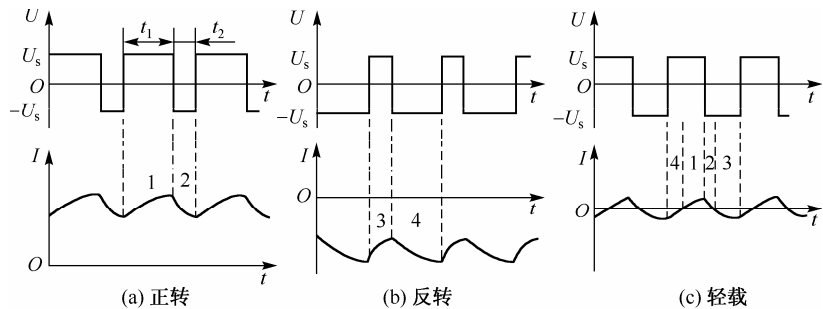


图 4.20 H 型双极可逆 PWM 电流波形

在双极性驱动下工作时，由于开关管自身有开关延时，并且“开”和“关”的延时时间不同，所以同一桥臂上的两个开关管容易出现直通现象，从而引起短路类的严重事故。为了防止直通，保证同一桥臂上的两个开关管在“开”与“关”的交替时，至少有一个开关管处于“关”的状态，通常在每一个 PWM 周期里，增加两个称为“死区”的低电平延时，死区的时间长短可根据开关管的种类及使用要求来确定，一般应在 $5\sim 20\ \mu\text{s}$ 。

单片机的专用 PWM 口大多没有死区设置功能；采用含有死区功能和驱动功能的专用集成电路（典型的直流电动机驱动专用芯片如 LMD18200 等），可以使小型直流电动机的控制电路非常简单。

2. 单极性驱动可逆 PWM 系统

单极性驱动是指在一个 PWM 周期内，电动机电枢只承受单极性的电压。其中最普遍的是受限单极性驱动方式和受限倍频单极性驱动方式。

(1) 受限单极性驱动可逆 PWM 系统

受限单极可逆 PWM 驱动系统与双极可逆系统的驱动电路相同，只是控制方式不同，如图 4.21 所示。

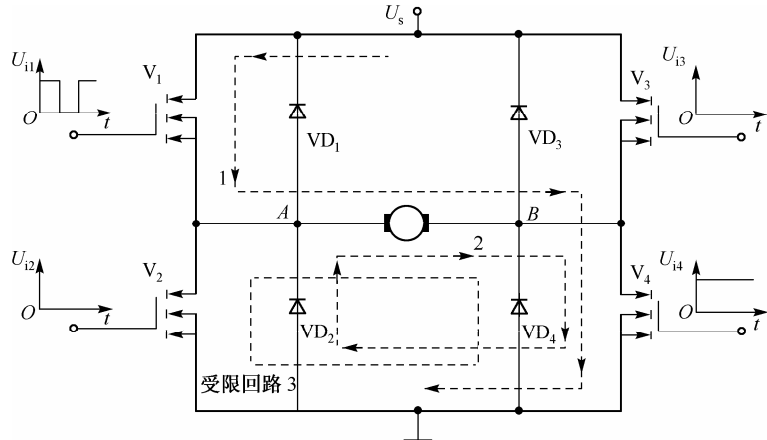


图 4.21 受限单极性可逆 PWM 驱动系统

要求电动机正转时，PWM 控制信号控制开关管 V_1 ，对开关管 V_4 加高电平使其常开，开关管 V_2, V_3 加低电平而截止（如图 4.21 中 U_i 所示）。

要求电动机反转时，PWM 控制信号控制开关管 V_3 ，对开关管 V_2 加高电平使其常开，开

关管 V_1, V_4 加低电平而截止。

电动机正转时，在每个 PWM 周期的 t_1 区间， V_1 导通，电流沿图 4.21 所示的虚线 1 从 A 到 B 流经电枢绕组，电动机工作在电动状态；在 t_2 区间， V_1 截止，电流在自感电动势的作用下，经 V_4 和 VD_2 形成续流回路如虚线 2，电动机继续工作在电动状态。正转时的电流波形如图 4.22(a) 所示，其电压、电流波形和占空比的计算都与不可逆 PWM 系统相同。

电动机制动时，PWM 控制信号的占空比减小，使电枢两端的平均电压小于反电动势。在反电动势的作用下，电流的路线应该是从 A 点出发，经 V_2, VD_4 到 B 来产生制动转矩，如图 4.21 的虚线 3。但是，由于 V_2 处于截止状态，使能耗制动电流通路受到限制，即所谓“受限”。

当电动机工作在轻载时，在每个 PWM 周期的 t_2 区间，当续流电流沿图 4.21 中虚线 2 流动并衰减到零后，由于 V_2 的截止使反电动势不能建立反向电流，电枢电流出现断流现象，如图 4.22(b) 所示。

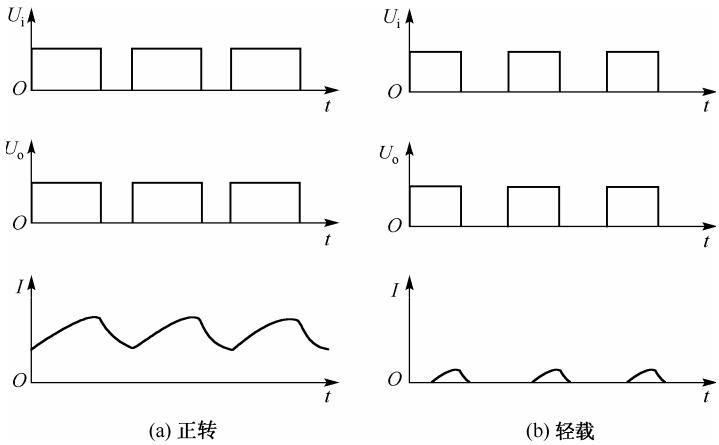


图 4.22 受限单极可逆 PWM 电流波形

受限单极性驱动方式的优点是，能够避免开关管直通，大大提高了系统的可靠性，适用于大功率、大转动惯量、可靠性要求较高的直流电动机控制；而其缺点是在轻载时会出现断流，克服方法是提高开关频率或改进电路设计。

(2) 受限倍频单极性驱动可逆 PWM 系统

受限倍频单极驱动方式通过改变对开关管的控制方式，使直流电动机电枢两端获得比 PWM 控制信号频率高 1 倍的电压波，以弥补受限单极驱动所产生的电流断流问题。

当要求电动机正转时，开关管 V_1, V_4 的 PWM 控制信号占空比和频率都相同，但相位相差 180° 。对另两个开关管 V_2, V_3 施加低电平，使它们始终截止，从而在电动机电枢两端产生两倍于 PWM 控制信号频率的“倍频”电压波形，如图 4.23(a) 所示。

当要求电动机反转时，占空比和频率相同而相位相差 180° 的 PWM 控制信号加在开关管 V_2, V_3 上，而开关管 V_1, V_4 始终截止。在电动机电枢两端产生的倍频电压波形如图 4.23(b) 所示。

受限倍频单极驱动方式的缺点是没有能耗制动能力，优点是电枢电压频率的提高，解决了断流问题，电枢电流的波动也减小了 $1/2$ ，因此多应用于大功率、可靠性较高的场合。

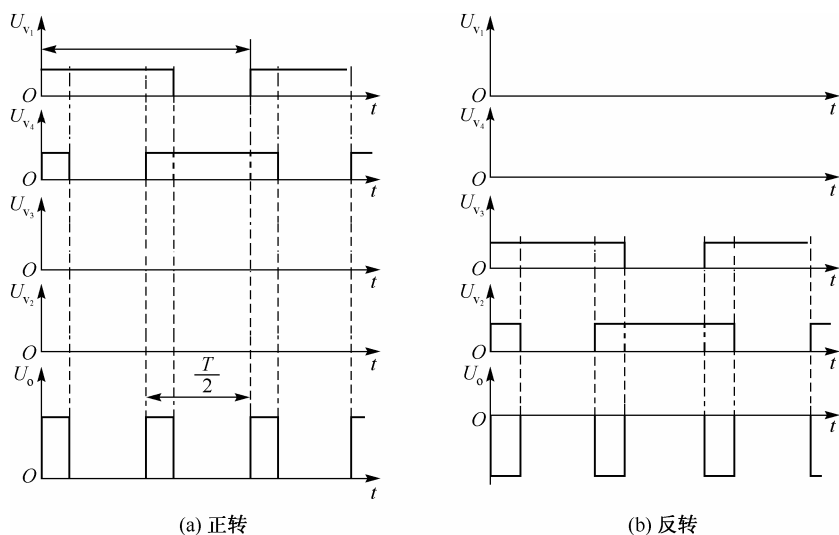


图 4.23 受限倍频单极可逆 PWM 电压波形

4.4.4 专用运动控制处理器

1. LM629 功能和工作原理

专用运动控制处理器 LM629 和微控制器、功率驱动器、直流电动机及增量式光电编码盘可以构成一个伺服系统，如图 4.24 所示。

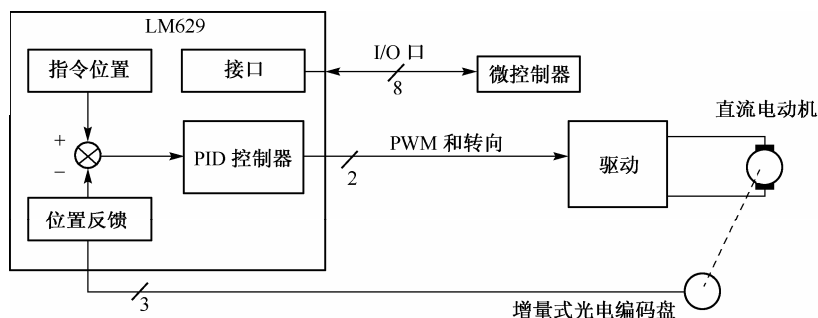


图 4.24 LM629 等构成的伺服系统框图

LM629 具有下列功能：

- 内置 32 位的位置、速度和加速度寄存器及 16 位可编程数字 PID 控制器，可编程微分项采样时间间隔；
- 8 位分辨率的 PWM 输出；
- 内部梯形速度图发生器，速度、位置和数字 PID 控制器等控制参数可以在控制过程中改变；
- 实时可编程中断，可对增量式光电编码盘的输出进行 4 倍频处理和信号处理。

LM629 通过 I/O 口与微控制器通信，输入运动参数和控制参数，输出状态和信息。

增量式光电编码盘用来反馈电动机的实际位置。来自增量式光电编码盘的位置信号 A, B ，经 LM629 四倍频，使分辨率提高。 A, B 逻辑状态每变化一次，LM629 内的位置寄存器就会加（减）1。当编码盘的 A, B, C 信号同时为低电平时，就产生一个 Index 信号，并将它送入 Index 寄存器，记录电动机的绝对位置。

LM629 的梯形速度图发生器用于计算所需的梯形速度分布图。在位置控制方式下，单片机送来加速度、最高转速、最终位置数据，LM629 利用这些数据计算运行轨迹，如图 4.25(a)所示。在电动机运行时，上述参数允许更改，如产生如图 4.25(b)所示的轨迹。在速度控制方式下，电动机用规定的加速度加速到规定的速度，并保持到执行新的速度指令。如果速度存在扰动，LM629 可使其平均速度恒定不变。

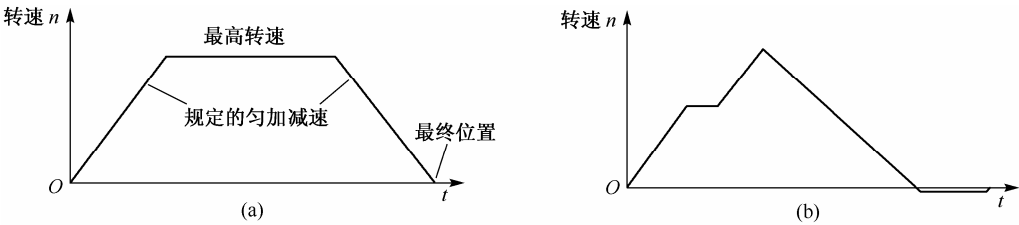


图 4.25 两种典型的速度轨迹

LM629 内部有一个数字 PID 控制器，用来控制闭环系统。数字 PID 控制器采用增量式 PID 控制算法，所需的 K_P , K_I , K_D 系数由单片机提供。

LM629 有 22 条指令，可用于单片机对其进行控制、数据传送和了解状态信息。指令可分成 5 类，包括初始化，中断，输入 PID 控制参数，输入位置、速度、加速度、控制方式和转向等控制参数，读状态和运动信息等，并且大多数指令可以在电动机运行过程中执行。

2. 典型应用

图4.26 所示的是采用 Intel 8751 单片机对 LM629 进行控制的位置伺服系统。LM629 的 I/O 口 D0~D7 与单片机的 P0 口相连，传送数据、控制指令和电动机的状态和运动信息。LM629 的中断引脚经一个非门与单片机的 \overline{INT}_0 相连，LM629 的 6 个中断源都通过该引脚申请单片机中断。当中断申请发生时，单片机通过读 LM629 的状态字来辨别中断源。

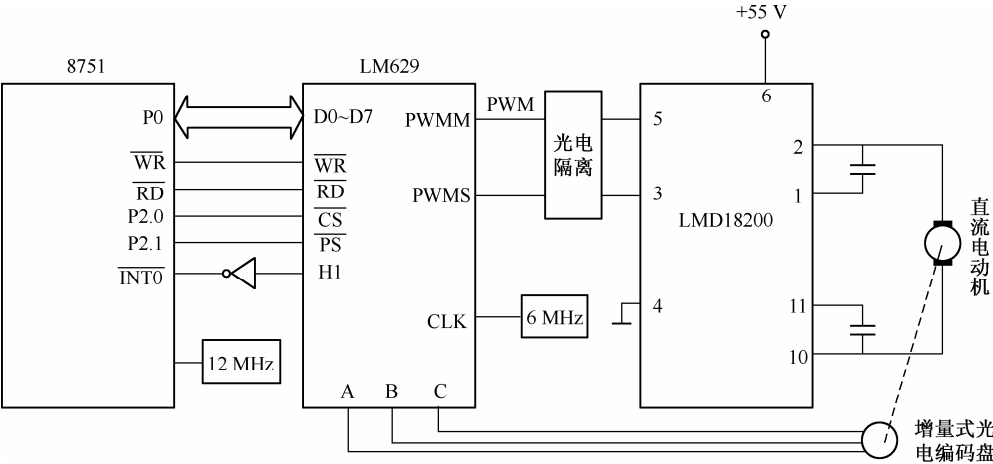


图 4.26 应用 LM629 组成的位置伺服系统

单片机的主要工作就是向 LM629 传送运动数据和 PID 数据，从而对电动机的运行进行监控。LM629 则根据单片机发来的数据生成速度图，进行位置跟踪，PID 控制和生成 PWM 信号输出。

LM629 的两个输出端 PWMS 和 PWMM 经光电隔离与驱动芯片 LMD18200 相连，用以驱动直流电动机运行。作为传感器的增量式光电编码盘安装在直流电动机输出轴上，其输出直接连到 LM629 的 A, B, C 输入端，形成反馈环节。

4.5 交流异步电动机控制

感应式交流异步电动机结构简单、体积小、重量轻、价格便宜、维护方便，是在生产和生活中应用最为广泛的一种电动机。

长期以来，交流异步电动机的调速是一个难题。直到 20 世纪 70 年代，由于微机控制技术和电力电子元件的发展，才使得交流异步电动机的调速成为可能，并得到迅速普及。常用的交流异步电动机调速控制有变频变压法（Variable Voltage Variable Frequency, VVVF）和矢量控制法等。

4.5.1 交流异步电动机变频调速原理

交流异步电动机的转速可由下式表示

$$n = \frac{60f}{p}(1 - s) \tag{4.7}$$

式中 n —— 电动机转速（转/分）；
 p —— 电动机磁极对数；
 f —— 电源频率；
 s —— 转差率。

在电动机的磁极对数 p 、转差率 s 和电源频率 f 等各项影响交流异步电动机转速的因素中，改变电源频率来实现电动机调速的效果最理想，这就是所谓变频调速。

1. 电路工作原理

变频调速的实质就是向交流异步电动机提供一个频率可控的电源。实现这一功能的装置称为变频器，由主电路和控制电路两部分组成，其中主电路通常采用交-直-交方式，即先将交流电转变成直流电（整流、滤波），再将直流电转变成频率可调的矩形波交流电（逆变）。常用的主电路原理图如图 4.27 所示。

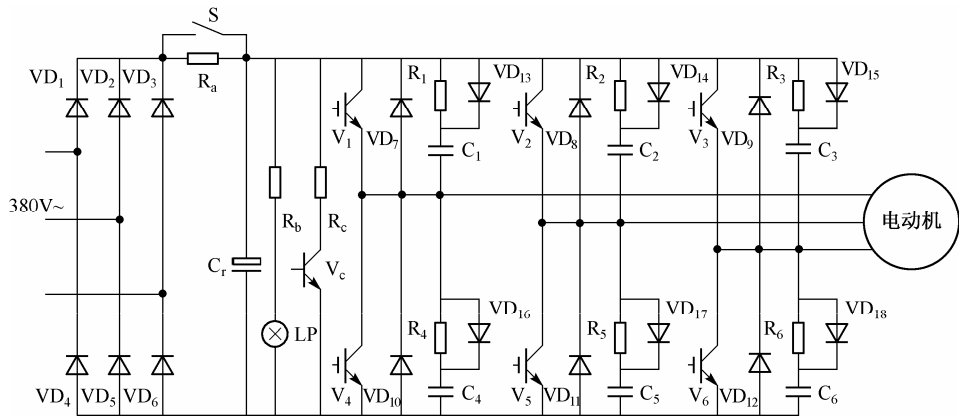


图 4.27 电压型交-直-交变频调速主电路

(1) 主电路

① 交-直电路。整流管 $VD_1 \sim VD_6$ 组成三相整流桥，对三相交流电进行全波整流。整流后的直流电压

$$U = 1.35 \times 380 \text{ V} = 513 \text{ V} \quad (4.8)$$

滤波电容 C_r 滤除整流后的电压波纹，并在负载变化时保持电压平稳。

当变频器通电时，瞬时冲击电流较大，为了保护电路元件，加限流电阻 R_a 延时一段时间，然后通过控制电路使开关 S 闭合，将限流电阻短路。

电源指示灯 LP 除了指示电源通断外，还在电源断开时，作为滤波电容 C_r 放电通路和指示。滤波电容 C_r 容量通常很大，放电的时间可能长达数分钟，故应避免高电压威胁人员安全。

制动电阻 R_c 为制动过程中处于发电状态的电动机提供一个放电通路，通过控制电路使开关管 V_c 导通，使再生电流消耗在电阻 R_c 上。

② 直-交电路。逆变开关管 $V_1 \sim V_6$ 组成三相逆变桥，将直流电逆变成频率可调的矩形波交流电。逆变管可以选择绝缘栅双极晶体管 IGBT 或功率场效应管 MOSFET。续流二极管 $VD_7 \sim VD_{12}$ 的作用是，当逆变开关管由导通状态变为截止时，为存储在电动机线圈中的电能提供释放通道；当电动机制动时，为再生电流提供回流到直流电源的通道。

③ 缓冲电路。保护逆变开关管的缓冲电路可以缓解开关管在开通和关断时，集电极电流 I_c 和集电极与发射极间电压 U_{CE} 的冲击（如图4.28所示）。当逆变开关管关断时，两端的并联电容 $C_1 \sim C_6$ 抑制 U_{CE} 电压的过高增长率。当逆变开关管开通时，电阻 $R_1 \sim R_6$ 限制电容的放电电流。为了避免电阻对电容充电的阻碍，在电阻两端还并联了二极管 $VD_{13} \sim VD_{18}$ 。这种缓冲电路的缺点是增加了损耗，因而只适用于中小功率变频器。

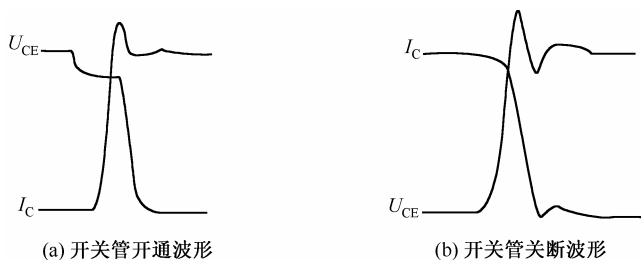


图 4.28 开关管开通与关断波形

缓冲电路的其他形式如图4.29所示，其中，图4.29(a)的交叉式缓冲电路避开了图4.27所示的缓冲电路的缺点，适用于中大功率变频器；图4.29(b)可吸收高于直流电压的电压尖峰，适用于小功率变频器；图4.29(c)在逆变开关管前面串联一个 di/dt 抑制电路，使缓冲效果更好。

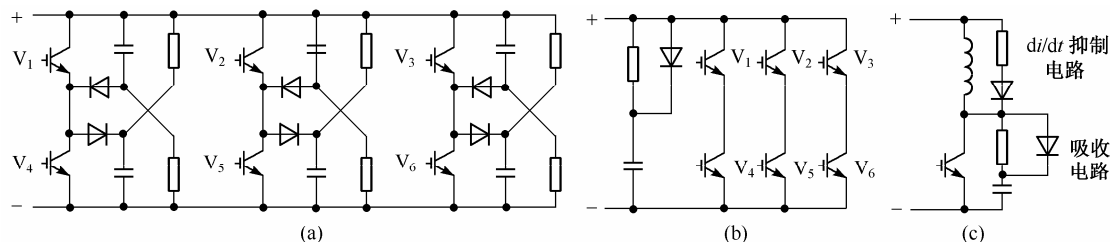


图 4.29 缓冲电路

(2) 三相逆变桥的工作原理

三相逆变桥的电路简图如图 4.30(a)所示，控制逆变管的导通和关断可以把直流电逆变成矩形波三相交流电，图中 R, Y, B 为逆变桥的输出。

180° 导通型三相逆变器的控制规律如图4.30(b)所示，其中深色部分表示逆变管导通。可以看出，每一时刻总有 3 个逆变管导通，另 3 个逆变管关断，并且 V_1 与 V_4 、 V_2 与 V_5 、 V_3 与 V_6 每对逆变管不能同时导通。线电压 U_{RY} 、 U_{YB} 、 U_{BR} 的波形如图 4.30(c)所示，可以看出，它们的幅值为 U ，三者之间互差 120°。各阶段的等值电路及相电压和线电压值如表 4.6 所示。

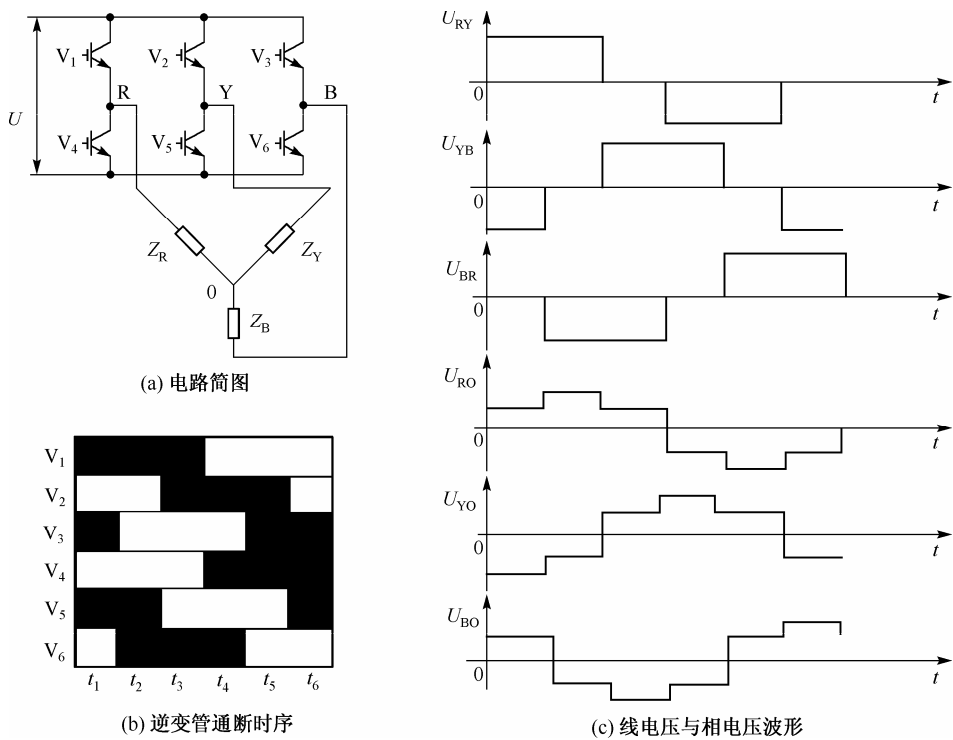


图 4.30 三相逆变桥工作原理与输出波形

表 4.6 180° 导通型三相逆变器各阶段的等值电路及相电压和线电压值

阶 段		0°~60°	60°~120°	120°~180°	180°~240°	240°~300°	300°~360°
导通管号		1, 3, 5	1, 5, 6	1, 2, 6	2, 4, 6	2, 3, 4	3, 4, 5
等值电路							
相 电 压	u_{RO}	$+\frac{1}{3}U_d$	$+\frac{2}{3}U_d$	$+\frac{1}{3}U_d$	$-\frac{1}{3}U_d$	$-\frac{2}{3}U_d$	$-\frac{1}{3}U_d$
	u_{YO}	$-\frac{2}{3}U_d$	$-\frac{1}{3}U_d$	$+\frac{1}{3}U_d$	$+\frac{2}{3}U_d$	$+\frac{1}{3}U_d$	$-\frac{1}{3}U_d$
	u_{BO}	$+\frac{1}{3}U_d$	$-\frac{1}{3}U_d$	$-\frac{2}{3}U_d$	$-\frac{1}{3}U_d$	$+\frac{1}{3}U_d$	$+\frac{2}{3}U_d$
线 电 压	u_{RY}	$+U_d$	$+U_d$	0	$-U_d$	$-U_d$	0
	u_{YB}	$-U_d$	0	$+U_d$	$+U_d$	0	$-U_d$
	u_{BR}	0	$-U_d$	$-U_d$	0	$+U_d$	$+U_d$

2. 变频与变压

交流异步电动机的定子绕组的反电动势是定子绕组切割旋转磁场磁力线的结果，其有效值可由下式计算：

$$E = Kf\Phi \quad (4.9)$$

式中 K ——与电动机结构有关的常数；

f ——电源频率；

Φ ——磁通。

而在电源一侧，电源电压的平衡方程式为

$$U = E + IR + jIX \quad (4.10)$$

加在电机绕组的电源电压 U ，一部分产生感应电动势 E ，另一部分消耗在阻抗（线圈电阻 R 和漏电感 X ）上。

定子电流 I 分成两部分： $I = I_1 + I_2$

式中， I_1 是少部分，用于建立主磁场磁通 Φ ； I_2 是大部分，用于产生电磁力带动机械负载。

交流异步电动机进行变频调速时，如频率 f 下降，则 E 降低。在电源电压 U 不变的情况下，定子电流 I 将增加。此时如果外负载不变，则 I_2 不变， I_1 将增加，也就是使磁通量增加。 Φ 的增加又使 E 增加，达到新的平衡点。

通常在设计时已使电动机的磁通容量达到最大容量，因此磁通量再增加将产生磁饱和，引起电流波形畸变，削弱电磁力矩，影响机械特性。

解决机械特性下降问题的一种方案是维持磁通量恒定不变，即设法使 $K\Phi = \text{常数}$ ，这就要求当电动机调速改变电源频率 f 时， E 也进行相应变化，以维持它们的比值不变。实际上，因为无法控制 E 的大小，通常是忽略在阻抗上产生的压降，用调节电源电压 U 来近似地代替调节 E ，使其跟随频率的变化，保持磁通量近似恒定，即

$$E/f \approx U/f = K\Phi = \text{常数} \quad (4.11)$$

这就是为什么在变频的同时也要变压，即 VVVF。

3. SPWM 调制波

用脉宽调制 PWM，可以方便地实现变频和变压：调节占空比 α ，就可以调节输出的平均电压；调节 PWM 波的频率 $1/T$ ，就可以改变电源频率，实现调速。

然而，矩形波含有许多高次谐波成分，将产生使交流异步电动机发热、力矩下降、振动噪声等不良后果。

使逆变电路输出的电压波形成为正弦波的一种方法是，将等宽的矩形波变成一组宽度渐变的脉冲波，其宽度变化规律应符合正弦的变化规律，如图 4.31 所示的正弦脉宽调制波，简称 SPWM 波。由于谐波成分大大减少，驱动效果可以达到基本满意的水平。

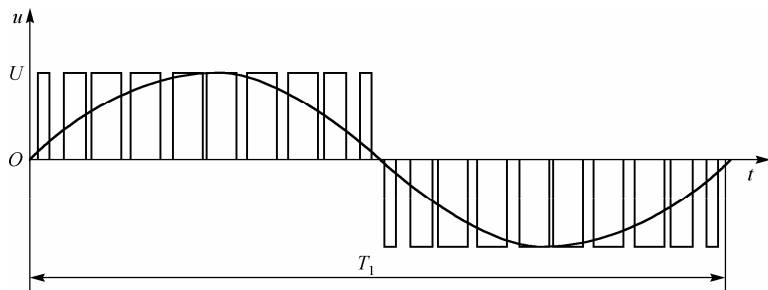


图 4.31 SPWM 波形

产生正弦脉宽调制波 SPWM 的通常方法是，用一组等腰三角形波与一个正弦波进行比较，

如图 4.32 所示，以其交点作为开关管“开”或“关”的时刻。等腰三角形波称为载波，而正弦波则称为调制波。改变正弦波的频率，就可以改变输出电源的频率，从而改变电动机的转速；改变正弦波的幅值，也就改变了正弦波与载波的交点，使输出脉冲系列的宽度发生变化，从而改变输出电压。

三相逆变开关管生成 SPWM 波的控制方式有单极性控制和双极性控制两种。

(1) 单极性控制

每半个周期内，在逆变桥的同一桥臂的上下两个逆变开关管中，只有一个逆变开关管按图 4.31 的规律反复通断，而另一个逆变开关管始终关断；在另外半个周期内，两个逆变开关管的工作状态正好相反。

三相逆变器中的 6 个逆变开关管的工作状态仍然可以用图 4.30(b) 进行描述，例如， V_1 开关管在 t_1, t_2, t_3 时间段中按 SPWM 波的规律进行开通和关断，在 t_4, t_5, t_6 时间段则全关断；同一桥臂的 V_4 开关管正好相反，在 t_1, t_2, t_3 时间段全关断，而在 t_4, t_5, t_6 时间段则按 SPWM 波的规律进行开通和关断；三个桥臂工作的规律都相同，只是在相位上相差 120° 。

(2) 双极性控制

在全部周期内，同一桥臂的上下两个逆变开关管交替开通与关断，形成互补的工作方式，其各种波形如图 4.33 所示。

图 4.33(a) 表示了三相调制波与等腰三角形载波的关系，三相调制波由频率和幅值都一样，但相位上相差 120° 的三条正弦波 u_R, u_Y, u_B 组成。每一条正弦波与等腰三角形载波的交点决定了同一桥臂（同相）逆变开关管的开通与关断时间。

图 4.33(b)、(c)、(d) 表示了各相电压 U_R, U_Y, U_B 输出的波形，它们的最大幅值是 $\pm U/2$ ，其中上臂开关管产生正脉冲，下臂开关管产生负脉冲。同样，三相相电压波形的相位也互差 120° 。

图 4.33(e) 是线电压 $U_{RY} = U_R - U_Y$ 的输出波形，同理也可以得到 $U_{YB} = U_Y - U_B, U_{BR} = U_B - U_R$ 。

4. 载波频率的选择

SPWM 波仅仅是近似正弦波，仍然含有高次谐波的成分。一般说来，载波频率应大于调制波频率 10~20 倍，载波频率越高，谐波波幅越小，电流波形越平滑。高的载波频率还可使变频器和电机的噪声进入超声范围，达到静音效果。

载波频率的提高，受逆变开关管的最高开关频率限制。采用 IGBT（工作频率可达几十 kHz 以上）为逆变开关管，可以得到较平滑的电流波形。

载波与调制波的频率调整有以下 3 种形式：

(1) 同步控制方式

使调制波频率与载波频率的比值等于常数，即在逆变器输出电压的每个周期内，所使用的三角波数目不变，因此产生的 SPWM 波的脉冲数是一定的。

这种控制方式的优点是，在调制波频率变化的范围内，逆变器输出波形的正负半波完全对称，使输出三相波形之间具有 120° 相差的对称关系。但在低频时有严重的不足：每个周期 SPWM 脉冲个数过少，使谐波分量加大。

(2) 异步控制方式

使载波频率固定不变，只调整调制波频率进行调速。虽然不存在同步控制方式所产生的

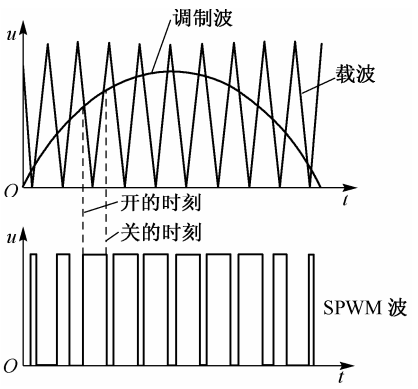


图 4.32 SPWM 波生成方法

低频谐波分量大的缺点，但是可能会造成逆变器输出的正半波与负半波、三相波之间不严格对称的现象，引起电动机运行不平稳。

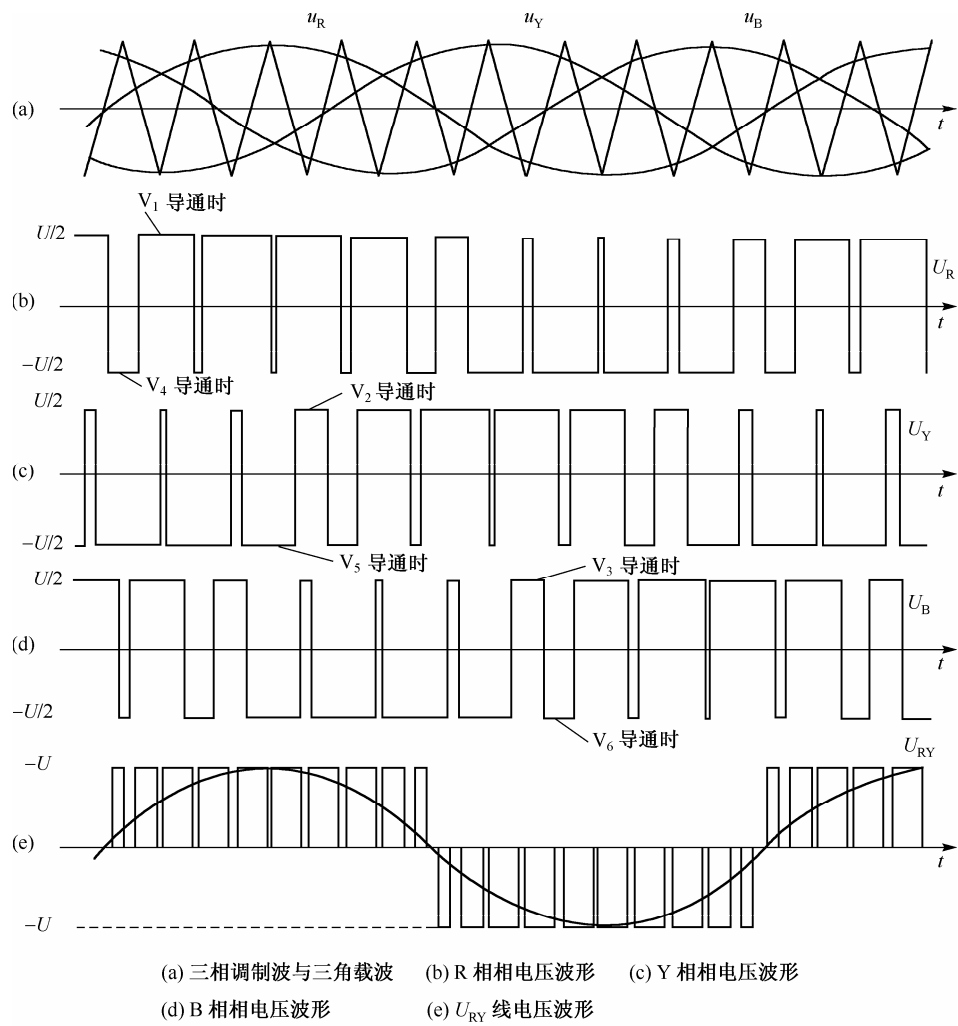


图 4.33 三相逆变器输出双极性 SPWM 波形图

(3) 分段同步控制方式

实际应用较多的分段同步控制方式综合了同步控制和异步控制的优点：在低频段使用异步控制方式，而在其他频率段使用同步控制方式。

4.5.2 变频调速的机械特性及其补偿

1. 变频调速电动机的机械特性

变频调速电动机的机械特性如图4.34所示。当电动机向低于额定转速方向调速时[如图4.34(a)所示]，曲线近似平行地下降，这说明，减速后的电动机仍然保持原来较硬的机械特性，但是随着电动机转速的下降，临界转矩减小，负载能力下降。

这是因为当频率 f 降得较低时，定子阻抗压降在电压 U 中所占的比例增大，已经不能满足 $U \approx E$ ，从而使得实际上 E/f 的比值减小，磁通量减小，导致电动机的临界转矩下降。

当电动机向高于额定转速方向调速时[如图 4.34(b)所示], 曲线不仅临界转矩下降, 而且曲线工作段的斜率开始增大, 使机械特性变软。

这是因为当频率 f 升高时, 电源电压受电动机绕组的绝缘强度限制, 不能相应地升高, 从而导致磁通量 Φ 下降。

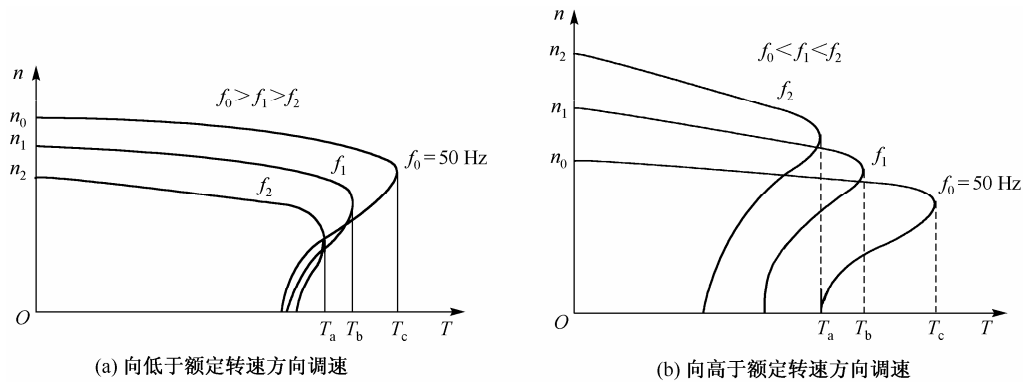


图 4.34 变频调速电动机的机械特性

2. V/F 转矩补偿法

通常采用 V/F 转矩补偿法, 也称为转矩提升 (Torque Boost) 法来解决在低于额定转速时调速使电动机负载能力减弱的问题。其原理是, 当频率 f 降低时, 适当提高电源电压 u 以保持磁通量恒定, 从而使电动机转矩回升。

提高电压 U 将使调压比 $K_U > K_f$, 也就是说, 电压 U 并不再随频率 f 等比例地变化, 而是按图 4.35 的曲线变化 (在实际的通用变频器中常有若干条简化曲线可供选择)。采用 V/F 转矩补偿法后的电动机机械特性如图 4.36 所示。

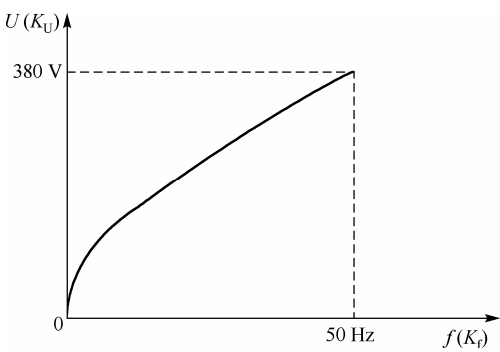


图 4.35 U/f 补偿曲线

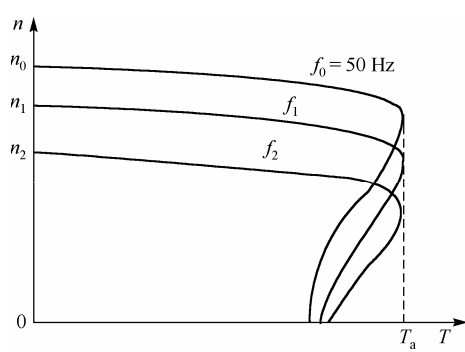


图 4.36 补偿后的机械特性

4.5.3 SPWM 波发生器及其应用

在 SPWM 波的频率、幅值和载波频率三项参数中, 任一项发生变化, 都将改变载波与调制波的交点, 因此, 每一次调整都需要重新计算坐标。采用专用芯片生成三相或单相 SPWM 波的控制信号, 可以方便地实现这些功能。

1. 工作原理

以 SA4828 为例, 专用于三相 SPWM 信号发生和控制的集成芯片的主要特点为全数字控

制，兼容 Intel 系列和 MOTOROLA 系列单片机，输出调制波频率范围 0~4 kHz，16 位调速分辨率，载波频率最高可达 24 kHz，内部 ROM 固化 3 种可选波形，可选最小脉宽和延迟时间（死区），可单独调整各相输出，以适应不平衡负载，有看门狗定时器。

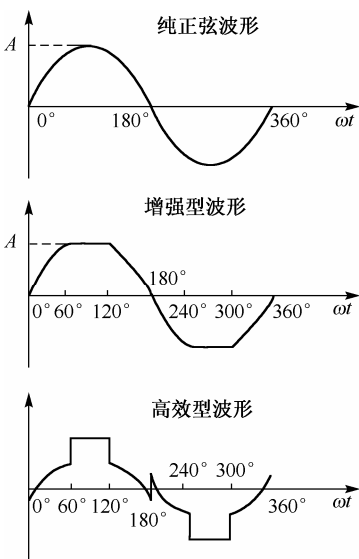


图 4.37 片内 ROM 存储的 3 种波形

在 SA4828 内部，来自 MCU 的数据通过总线控制和译码进入初始化寄存器或控制寄存器，控制相控逻辑电路。外部时钟输入经分频器分成设定的频率，并生成三角形载波，与所选定的片内 ROM 中的调制波形进行比较，自动生成 SPWM 输出脉冲。通过脉冲删除电路，删去只会增加开关管损耗的过窄脉冲。通过脉冲延迟电路生成死区，保证任何桥臂上的两个开关管不会在状态转换期间短路。看门狗定时器用来防止程序跑飞，一旦程序运行超时立即快速封锁输出。

片内 ROM 存储有纯正弦、增强型和高效型 3 种可供选择的波形（如图 4.37 所示），每一种波形各 1536 个采样值。增强型波形又称为三次谐波，它可以使输出功率提高 20%，三相谐波互相抵消，防止电动机发热。高效型波形又称为带死区的三次谐波，它是进一步优化的三次谐波，可以减小逆变开关管的损耗，提高功率利用率。

2. 典型应用

SA4828 与 Intel8751 单片机连接构成三相交流异步电动机调速系统的电路图，如图 4.38 所示。

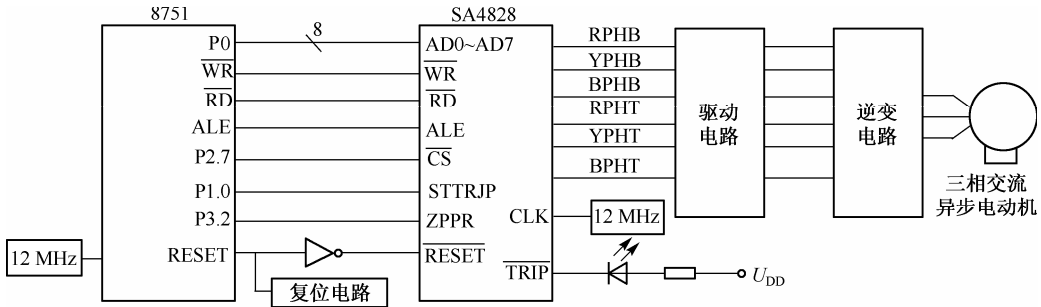


图 4.38 SA4828 与 8751 构成三相交流异步电动机调速系统

SA4828 的 STTRIP 引脚接 8751 的 P1.0，使单片机能够在异常情况下封锁 SA4828 的输出。ZPPR 引脚接 8751 的 P3.2（INT0），测量调制波的频率，用于显示。TRIP 引脚接一个发光二极管，当 SA4828 的输出被封锁时，发光二极管亮，用来指示封锁状态。6 个输出引脚 RPHT, YPHT, BPHT, RPHB, YPHB, BPHB 分别通过各自的驱动电路，来驱动逆变桥的 6 个开关管。

4.5.4 矢量控制与直接转矩控制

1. 旋转磁场分析

对空间均匀分布的任意多相绕组通以时间上均匀分布的多相平衡电流，就可以产生旋转磁场。例如：

(1) 三相旋转磁场

对空间上相差 120° 的三相绕组 A, B, C 通入相位上各相差 120° 的三相交流电 i_A, i_B, i_C 后, 产生一个合成磁感应强度不变的旋转磁场。电流交变一个周期, 合成磁场的轴线也在空间旋转一周。

三相交流电动机的旋转磁场就是这样产生的。

(2) 两相旋转磁场

对空间上相差 90° 的两相固定绕组 α, β 通入相位上相差 90° 的两相平衡的交流电流 i_α, i_β (如图 4.39 所示), 其合成磁场如图 4.40 所示。可以看出, 两相合成磁场也具有和三相旋转磁场完全相同的特点。

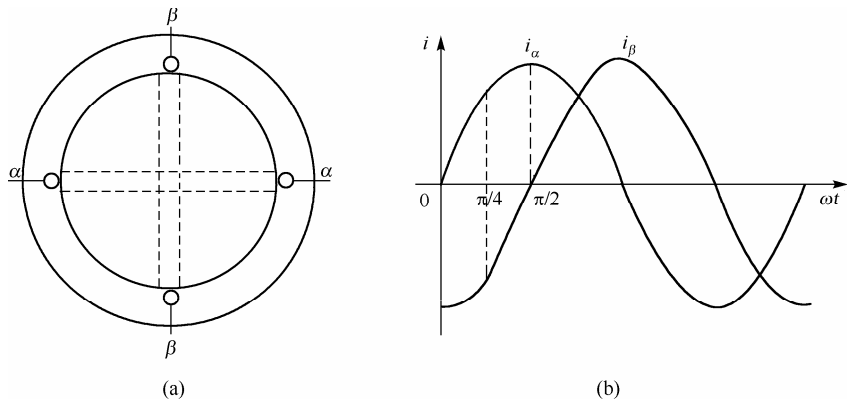


图 4.39 两相绕组和两相交流电流

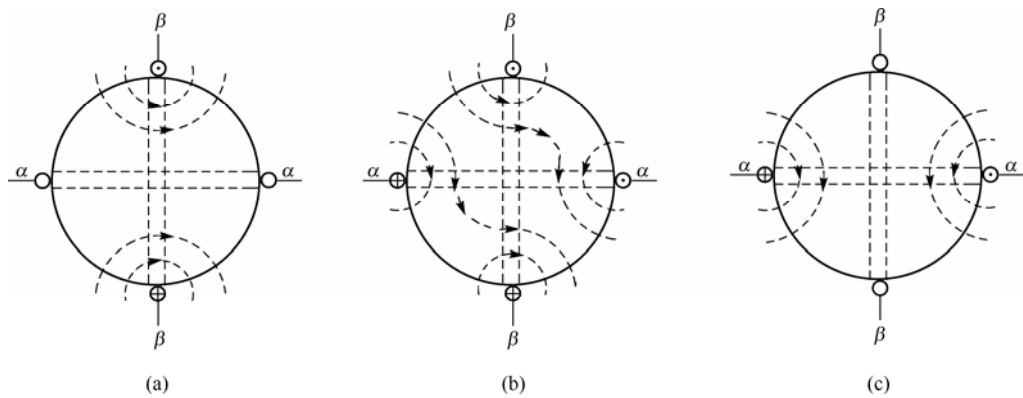


图 4.40 两相合成磁场

(3) 旋转体的旋转磁场

在旋转体上, 放置一个通入直流电流的绕组 M , 将产生一个恒定磁场。当旋转体旋转时, 恒定磁场也随之旋转, 在空间形成了一个旋转磁场。

如果在旋转体上放置两个互相垂直的直流绕组 M, T , 则当给这两个绕组分别通入直流电流时, 它们的合成磁场仍然是恒定磁场, 如图 4.41 所示。同样, 当旋转体旋转时, 该合成磁场也随之旋转, 称为机械旋转直流合成磁场。而且, 调节直流电流 i_M 或 i_T , 都可以调整直流合成磁场的强度。

如果用上述 3 种方法产生的旋转磁场的磁极对数、磁感应强度和转速都完全相同, 则认为这时的三相磁场、两相磁场和旋转直流磁场是等效的, 可以互相进行等效转换。

通常，把三相系统向两相系统的转换称为 3/2 变换，两相系统向三相系统的转换称为 2/3 变换，把两相交流系统向旋转直流系统的转换称为交/直变换，旋转直流系统向两相交流系统的转换称为直/交变换。

2. 矢量控制的基本思想

为了使交流异步电动机调速后的机械特性及动态性能都能够像直流电动机那样优良，交流电动机应当也能像直流电动机那样，对负载电流和励磁电流分别进行独立的控制，并使它们的磁场在空间位置上也互差 90° 、独立地进行调节。矢量控制（Vector Control, VC）将异步电机在三相静止的 ABC 坐标系上的数学模型，转换为两相同步旋转坐标系按转子磁场定向（Magnetization Torque, MT）的数学模型。坐标系的变换（相当于矩阵变换）实现了电机定子电流两个分量之间的解耦，从而可以求出：定子电流的励磁分量 i_{m1} 唯一决定转子磁链 Ψ_2 ，定子电流的转矩分量 i_{m2} 唯一决定电机的转矩 T_e ，使 i_{m1} 和 i_{m2} 与直流电机中的励磁电流和电枢电流相对应，从而大大简化了多变量强耦合的交流变频调速系统的控制问题。

如果将控制交流调速的给定信号变换成类似于直流电动机磁场系统的控制信号，也就是说，假想由两个互相垂直的直流绕组同处于一个旋转体上，两个绕组中分别独立地通入由给定信号分解而得的励磁电流信号 i_M 和转矩电流信号 i_T ，则通过等效变换，可以得到与这两个基本控制信号 i_M 、 i_T 等效的三相交流控制信号 i_A 、 i_B 、 i_C 来控制逆变电路。同样，对于电动机在运行过程中系统的三相交流数据，也可以等效变换成两个互相垂直的直流信号，反馈到控制端，用来修正基本控制信号 i_M 、 i_T 。

在进行控制时，可以和直流电动机一样，使其中一个磁场电流（如 i_M ）不变，而控制另一个磁场电流（如 i_T ）信号，从而获得和直流电动机类似的控制效果。

矢量控制的基本原理框图如图 4.42 所示，给定信号分解成两个互相垂直且独立的直流信号 i_M 、 i_T ，然后通过直/交变换将 i_M 、 i_T 变换成两相交流信号 i_α 、 i_β ，又经 2/3 变换，得到三相交流的控制信号 i_A 、 i_B 、 i_C 来控制逆变电路。

电流反馈信号经 3/2 变换和交/直变换，传送到控制端，分别修正直流控制信号的磁场分量 i_M 和转矩分量 i_T ，从而模拟出类似于直流电动机的工作状况。

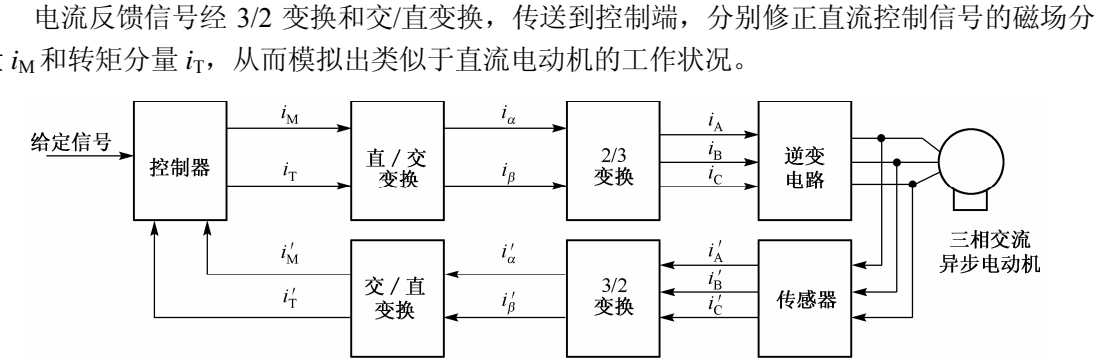


图 4.42 矢量控制原理框图

3. 直接转矩控制（DTC）

采用直接转矩控制的交流调速系统可以获得比矢量控制快得多的转矩响应。直接转矩控制把转矩直接作为被控量，以转矩为中心来进行磁链、转矩的综合控制。通过简单地检测电

机定子电压和电流，采用定子磁链控制磁场定向，用离散的电压状态和六边形磁链轨迹或近似圆形磁链轨迹的概念，不需要旋转坐标变换，直接在定子坐标系下分析异步电动机的数学模型，计算与控制异步电动机的定子磁链和转矩。采用离散的两点式调节器（Band-Band 控制），比较转矩检测值与转矩给定值，使转矩波动限制在一定的容差范围内。容差的大小由频率调节器来控制，并产生 PWM 脉宽调制信号，直接对逆变器的开关状态进行控制，可以获得高动态性能的转矩输出，但其开关频率是随机变化的。

直接转矩控制的控制结构简单、控制信号处理的物理概念明确。由于没有电流控制环路，Band-Band 控制产生的输出电压，不存在电流调节的时间滞后，也没有任何电流限制，电压可以出现过冲现象，故电机可以获得较大的 du/dt ，较大的加速电流，从而产生较快的电流响应及转矩响应。

4. 直接转矩控制与矢量控制的比较

DTC 变频器可以获得较好的转矩响应，但稳态指标较差。适用于交流伺服传动、机车牵引等对转矩响应要求高的场合。而一些对转矩响应要求不苛刻，特别是带有齿轮连接的传动，过快的转矩响应不仅不利，反而有害。

VC 变频器在可以确定的开关频率条件下，采用消除谐波的 PWM 控制方法，输出电压、电流的谐波较小。在相同电力电子元件条件下，变频器的容量利用率和效率较高。对于那些不求较高动态性能指标，如风机、水泵节能传动和一般工业机械传动等应用场合，更为有利。

DTC 与 VC 采用同样的交流电机数学模型，DTC 的低速特性可以借助于 VC 得到改善。DTC 和 VC 控制系统都需要速度信息，无速度传感器控制的产品化是交流电机调速控制的重要课题。

4.6 步进电动机控制

步进电动机可以将电脉冲信号直接转变成角位移，非常适合于微型计算机控制，其优点如下：

- 角位移与输入脉冲数严格成正比，跟随性好，没有累计误差；
- 可以与驱动电路组成简单、廉价、可靠的开环数控系统，也可以与角度反馈环节组成高性能的闭环数控系统；
- 动态响应快，易于启停、正反转及变速；
- 可在相当宽的范围内平滑调节速度，低速下仍能保证获得大转矩，一般不用减速器就可以直接驱动负载。

步进电动机的缺点如下：

- 不能直接使用交流电源和直流电源，只有通过脉冲电源供电才能运行；
- 存在振荡和失步现象，必须对控制系统和机械负载采取相应的措施；
- 自身的噪声和振动较大，带惯性负载的能力较差。

4.6.1 反应式步进电动机的结构和工作原理

1. 步进电动机的分类

步进电动机可分为以下 3 类。

(1) 反应式（Variable Reluctance, VR）

转子用软磁材料制成，其中没有绕组。结构简单、成本低，步距角可以做得很小，但动态性能较差。

(2) 永磁式 (Permanent Magnet, PM)

转子用永磁材料制成, 本身就是一个磁源。输出转矩大, 动态性能好, 转子的极数与定子的极数相同, 一般步距角较大, 需供给正负脉冲信号。

(3) 混合式 (Hybrid, HB)

混合式步进电动机综合了反应式和永磁式两者的优点, 它的输出转矩大, 动态性能好, 步距角小, 但结构复杂, 成本较高。

2. 反应式步进电动机的结构

反应式步进电动机的性能价格比高, 应用广泛。三相反应式步进电动机结构如图 4.42 所示。定子由硅钢片叠成, 每相一对磁极, 每个磁极的内表面都分布着多个小齿, 它们大小相同, 间距相同。

转子是由软磁材料制成的, 其外表面也均匀分布着小齿, 这些小齿与定子磁极上的小齿的齿距相同, 形状相似。

定子和转子的齿距角 θ_Z 都等于

$$\theta_Z = 2\pi/Z \quad (4.12)$$

式中, Z 是转子的齿数。

例如, 如果转子的齿数为 40, 则齿距角为

$$\theta_Z = 2\pi/40 = 9^\circ$$

反应式步进电动机对空间结构的要求是, 定子绕组磁极的分度角 (如三相的 120° 和 240°) 不能被齿距角整除。定子绕组磁极的分度角被齿距角除后所得的余数, 是步距角的倍数, 而且倍数与相数没有公因子。

定子磁极小齿与转子小齿对齐的状态称为对齿, 定子磁极小齿与转子小齿不对齐的状态称为错齿。步进电动机的空间结构设计决定了当某一相定子磁极与转子形成对齿时, 其他相都处于错齿状态。

3. 反应式步进电动机的步进原理

对齿和错齿交替的空间分布与通电时序相配合是步进电动机旋转的条件。若给处于错齿状态的相通电, 则转子将在电磁力的作用下, 向磁阻最小的位置, 即通电相趋于对齿的位置转动。

如图 4.43 所示, 当开关 S_A 合上时, A 相绕组通电, 建立 A 相磁场使 A 相定子磁极与转子形成对齿; B、C 相则处于错齿状态。

若齿距角为 9° , 将 A 相磁极中心线看成 0° , 此处的转子齿为 0 号齿, 则 120° 处的 B 相磁极中心线位于转子第 13 号齿再过 $1/3$ 齿距角, 即 B 相错齿 3° , 而 240° 处的 C 相磁极中心线位于转子第 26 号齿再过 $2/3$ 齿距角, 即 C 相错齿 6° 。

下一步将 A 相断电, 同时给处于错齿状态的相 (例如, B 相) 通电, 则转子将向磁阻最小的位置 (B 相对齿) 转动。结果是转子转动了 $1/3$ 个齿距角, B 相对齿、C 相错齿 $1/3$ 个齿距角和 A 相错齿 $2/3$ 个齿距角。

这样, 按 A—B—C—A 顺序轮流给各相绕组通电, 磁场将按 A—B—C—A 方向转过 360° , 转子则沿同一方向转过一个齿距角。显然, 改变通电顺序也就改变了转子的转向。

绕组通电一次的操作称为一拍; 按 A—B—C—A 顺序, 三相反应式步进电动机磁场旋转一周需要三拍。

转子每拍走一步, 按 A—B—C—A 顺序, 转一个齿距角需要三步。

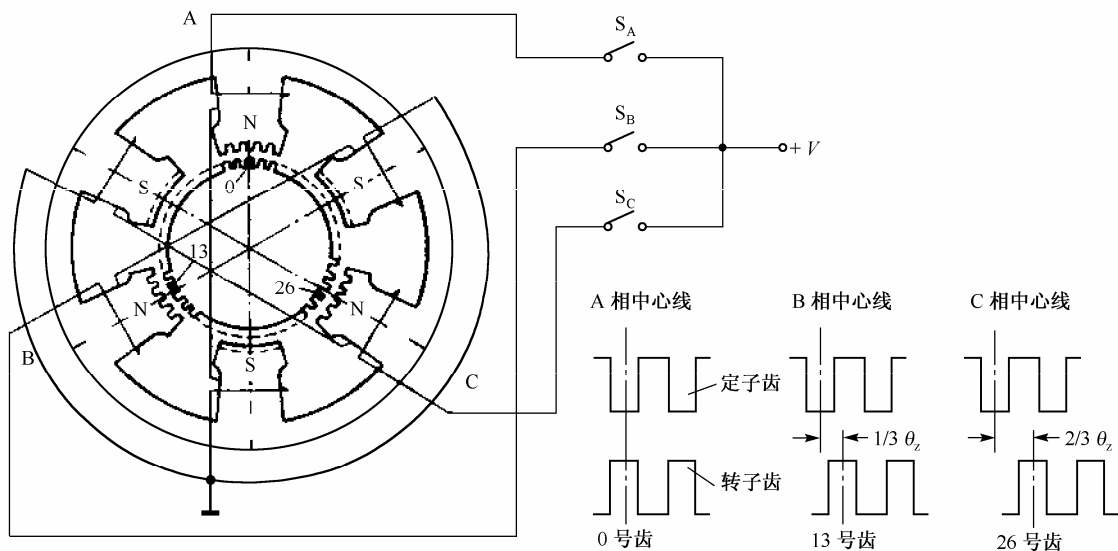


图 4.43 三相反应式步进电动机结构

转子走一步所转过的角度称为步距角，计算公式如下：

$$\theta_N = \theta_Z / N = 2\pi / NZ \quad (4.13)$$

式中 N ——步进电动机工作拍数。

4.6.2 步进电动机的工作方式

1. 单三拍工作方式

三相步进电动机按 A—B—C—A 方式循环通电的工作方式称为单三拍工作方式。其中“单”指的是每次对一个相通电，“三拍”指的是磁场旋转一周需要换相三次，这时转子转动一个齿距角。如果 N 相步进电动机每次只对一相通电，则磁场旋转一周需要 N 拍。

单三拍工作方式的各相通电波形如图 4.44 所示。电压波形是方波，由于存在着绕组电感，电流波形如虚线所示由两段指数曲线组成。为了使绕组电流也能快速变化，在通电时迅速建立磁场，断电时不干扰其他相磁场，可以在续流二极管回路中串联一个电阻，促使电流波形快速下降。

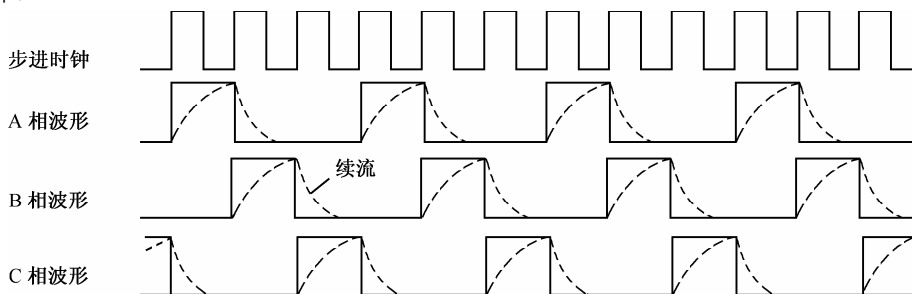


图 4.44 单三拍工作方式时的相电压、电流波形

2. 双三拍工作方式

双三拍的“双”是指每次对两相同时通电。“三拍”的含义与单三拍一样，即换相三次，磁场旋转一周，转子转动一个齿距角。如果步进电动机正转的通电顺序为 AB—BC—CA，则反转通电顺序为 AB—CA—BC。

各相通电的波形如图 4.45 所示。每一拍中，都有两相通电，每一相通电时间都持续两拍，所以通电的时间长，消耗的电功率大，当然，获得的电磁转矩也大。

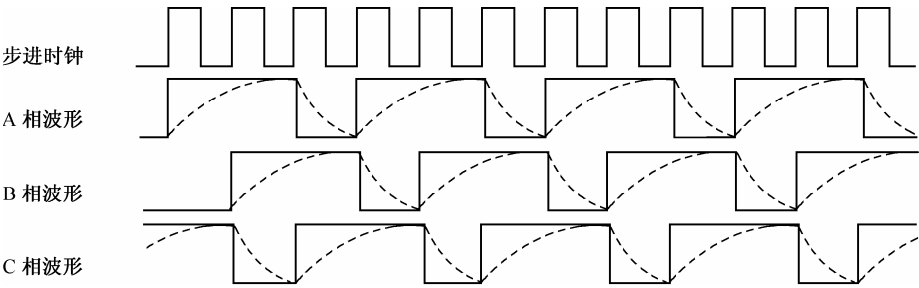


图 4.45 双三拍工作方式时的相电压、电流波形

与单三拍不同，双三拍工作时的磁阻最小位置没有一相磁极与转子对齿。当 AB 两相通电时，最小磁阻位置是转子齿与 A、B 两相磁极的齿分别错开 $\pm 1/6$ 个齿距角的位置，此时转子齿与 C 相错 $1/2$ 个齿距角。同样，当 BC 两相通电时，最小磁阻位置是转子齿与 B、C 两相磁极的齿分别错 $\pm 1/6$ 个齿距角的位置；当 CA 两相通电时，最小磁阻位置是转子齿与 C、A 两相磁极的齿分别错 $\pm 1/6$ 个齿距角的位置。

双三拍方式不易产生失步，有利于在低频区工作。这是因为当两相通电后，其中一相产生的电磁力起了阻尼作用。绕组中电流越大，阻尼作用就越大。而单三拍是单相通电励磁，不会产生阻尼作用，工作在低频区时通电时间长而使能量过大，易产生过冲现象。

3. 六拍工作方式
六拍工作方式是单三拍与双三拍交替使用的一种方法，也称为单双六拍或 1-2 相励磁法。各相通电的电压和电流波形如图 4.46 所示。

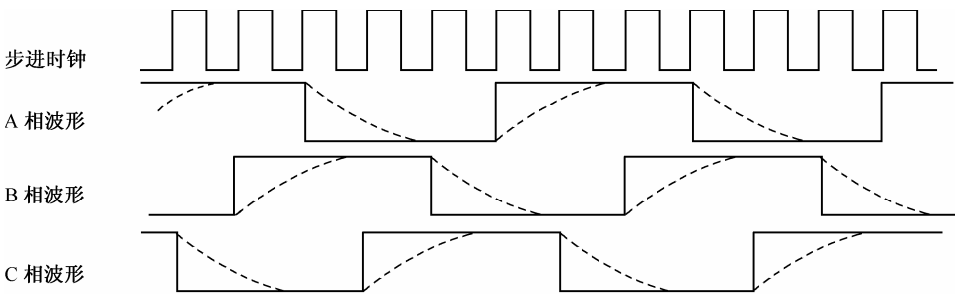


图 4.46 六拍工作方式时的相电压、电流波形

步进电动机的正转通电顺序为 A—AB—B—BC—C—CA；反转通电顺序为 A—CA—C—BC—B—AB。也就是说，磁场旋转一周，通电需要换相六次（即六拍）。由于转子转动一个齿距角需要六拍，此时的步距角比单三拍和双三拍时小一半，所以步进精度高一倍。

在步进电动机控制的应用中，选择合适的工作方式非常重要。单三拍、双三拍、六拍这 3 种工作方式的比较如表 4.7 所示。一般来说，六拍工作方式的性能较好，单三拍工作方式的性能较差。

多相步进电动机，可以有多种工作方式，例如，四相步进电动机，有单四拍（A—B—

C—D)、双四拍 (AB—BC—CD—DA)、八拍 (A—AB—B—BC—C—CD—D—DA，或者 AB—ABC—BC—BCD—CD—CDA—DA—DAB) 等。

表 4.7 3 种工作方式的比较

工 作 方 式	单 三 拍	双 三 拍	六 拍
步进周期	T	T	T
每相通电时间	T	$2T$	$3T$
走齿周期	$3T$	$3T$	$6T$
相电流	小	较大	最大
高频性能	差	较好	较好
转矩	小	中	大
电磁阻尼	小	较大	较大
振荡	易	较易	不易
功耗	小	大	中

4.6.3 步进电动机特性

1. 步进电动机的振荡、失步及解决方法

从能量供给的角度来看，能量供给过度会使步进电机的运动产生振荡，供给不足则会产生失步。

(1) 振荡

步进电动机工作在低频区、共振区或突然停止时，会产生振荡现象。

当步进电动机工作在低频区时，由于励磁脉冲间隔的时间较长，步进电动机表现为单步运行，转子围绕平衡点形成振荡。在机械摩擦和电磁阻尼的作用下，振荡逐渐衰减，最终稳定在平衡点。

当步进电动机工作在共振区时，脉冲频率与步进电动机的振荡频率或它的分频、倍频接近，从而使振荡加剧，严重时造成失步。在第 1 个脉冲到来后，转子经历了一次对平衡点的过冲和回摆振荡。如果当转子回摆到最大幅值时，恰好第 2 个脉冲到来，则转子受到的电磁转矩为负值，使转子继续回摆。接着第 3 个脉冲到来，转子受正转矩的作用回到平衡点。这样，转子经过 3 个脉冲仍然回到原来位置，也就是丢了 3 步。

当步进电动机工作在高频区时不会产生振荡，这是由于换相周期短，转子来不及反冲。同时，绕组中的电流尚未上升到稳定值，转子没有获得足够的能量。

减小步距角可以削弱振荡，减小振荡幅值。

(2) 失步

步进电动机的失步就是转速不受步进脉冲的控制，有两种情况。

① 转子的转速慢于旋转磁场的速度，或者说慢于换相速度。步进电动机有一个启动频率，超过该频率启动时，肯定会产生失步。这是因为，如果脉冲的频率较高，步进电动机在启动时来不及获得足够的能量，转子就无法跟上旋转磁场的速度，引起失步。但启动频率并不是一个固定值，提高电动机的转矩、减小负载转动惯量、减小步距角都可以提高步进电动机的启动频率。

② 转子的平均速度大于旋转磁场的速度。主要发生在制动和突然换向时，转子获得过多的能量，产生严重的过冲，引起失步。

(3) 阻尼方法

增加阻尼可以消除振荡，主要有机械阻尼法和电子阻尼法。

机械阻尼法就是在电动机轴上加阻尼器，电子阻尼法则有多种。

① 多相励磁法。采用多相励磁，例如，三相步进电动机的双三拍和六拍方式，可以产生电磁阻尼，削弱或消除振荡现象。

② 变频变压法。低频时，绕组中的电流上升时间长，转子获得的能量大，容易产生振荡，高频时则相反。如果随频率的降低使电压减小，则低频时绕组的电流减小，即减少了转子获得的能量，可以有效地消除振荡。

③ 细分步法。细分步法是将步进电动机绕组中的稳定电流分成若干阶级，每进一步时电流升一级，同时，也相对地提高步进频率，使步进过程平稳进行。

④ 反相阻尼法。这种方法用于步进电动机制动。在步进电动机转子要过平衡点之前，加一个反向作用力去平衡惯性力，使转子到达平衡点时速度为零，实现准确制动。例如，三相步进电动机工作在单三拍，目前处于 B 拍，并希望它停在 C 拍，则控制换相为 B—C—B—C。第 2 个 B 拍的作用是平衡惯性力。

2. 步进电动机的矩角特性

转子处于平衡点的位置称为零位，转子偏离零位线的夹角称为失调角 θ_e ，转子偏离零位后产生的电磁转矩与失调角 θ_e 之间的关系就称为步进电动机的矩角特性。

(1) 单相通电

当失调角 $\theta_e=0$ 时，电磁力在转子的切线方向上没有分力，转子不转动。

如果转子偏离零位，则电磁力在转子切线方向的分力形成转矩。随着失调角的增加（顺时针为正），产生的转矩增大。当 $\theta_e=\theta_z/4$ 时，转矩最大，但转矩的方向是逆时针的负转矩。

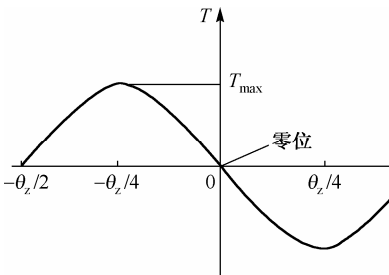


图 4.47 步进电动机的矩角特性

步进电动机的转矩随失调角的这种变化规律曲线近似正弦曲线，如图 4.47 所示。其中，最大转矩 T_{max} 表示了步进电动机承受负载的能力，是步进电动机最主要的性能指标之一。

(2) 多相通电

根据叠加原理，多相通电时的矩角特性可近似地由每相单独通电时的矩角特性叠加求出。三相步进电动机双三拍方式的矩角特性如图 4.48 所示，A、B 两相单独通电时两条曲线相位差 $\theta_z/3$ ，最大转矩分别为 T_A 、 T_B ；A、B 两相的矩角特性曲线叠加，就可以得到 AB 同时通电时的矩角特性曲线，其最大转矩为 T_{AB} 。

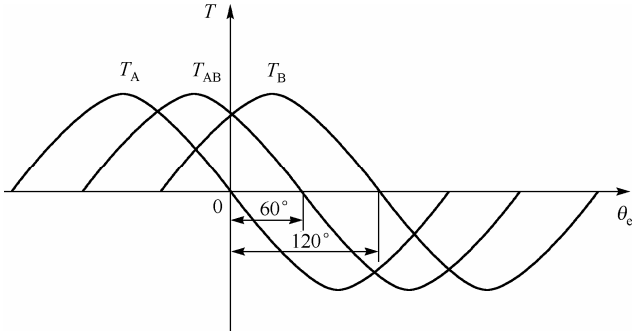


图 4.48 三相步进电动机单相、双相通电时的矩角特性

由图可见，对于三相步进电动机来说，两相通电时的最大转矩与单相通电时的最大转矩相同，也就是说，增加通电相数并不能提高最大转矩。

(3) 最大负载转矩与步距角

以三相步进电动机单三拍方式为例，来分析随失调角变化的转矩是怎样驱动负载的。

三相步进电动机单三拍方式时运行的矩角特性如图 4.49 所示。A 相通电时转矩随失调角的变化，如曲线 A 所示。切换为 B 相绕组通电，转矩的变化规律也切换为曲线 B。可以看出，步进运行能提供的最大连续驱动负载转矩为 T_q ，即 A、B 两条曲线的交点纵坐标。只有当负载转矩 $T_L < T_q$ 时，步进电动机才能带动负载进行步进运动，因此， T_q 被称为最大负载转矩或启动转矩。

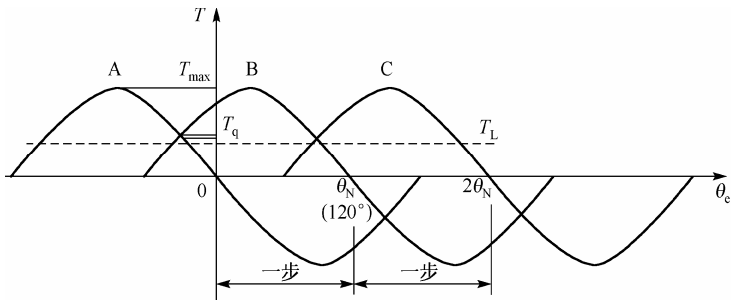


图 4.49 三相步进电动机单三拍运行时的矩角特性

步距角 θ_N 越小，A、B 两条曲线的交点越上移， T_q 越接近 T_{max} ，步进电动机负载能力越大。所以，减小步距角有利于提高步进电动机的负载能力。

3. 步进电动机的矩频特性

步进电动机的输出转矩与控制脉冲频率之间的关系称为矩频特性。一般是如图 4.50 所示的一条下降曲线。输出转矩以最大负载转矩（启动转矩） T_q 为起点，随着控制脉冲频率 f 的增加而下降。

由于绕组电感和驱动电路电阻（包括绕组线圈电阻、限流电阻和晶体管结电阻）的存在，绕组中的电流上升和下降都需要一定时间。当脉冲频率较低时，绕组中通电的周期较长，电流的平均值较大，电动机获得的能量较高，可以维持较高的转矩；当脉冲频率较高时，电动机获得的能量较少，因此转矩下降。

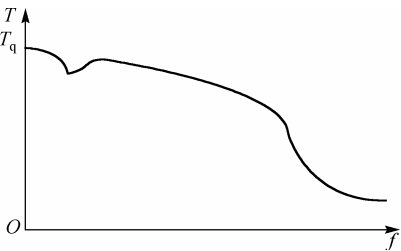


图 4.50 步进电动机的矩频特性

随着频率上升，转子转速升高，在定子绕组中产生的附加旋转电势使电动机受到更大的阻尼转矩。铁心的涡流损失增加，也使步进电动机输出转矩下降。

矩频特性曲线上的凹陷可看成是步进电动机的共振区，由于共振消耗一定的能量，使转矩下降。

提高矩频特性的高频性能的方法有：

(1) 减小驱动电路的时间常数

增加驱动电路电阻可以减小时间常数，但会使通电回路中电流值减小。为了保证通电回路中的电流不变，应在增加电阻的同时提高电源电压。

(2) 采用多相励磁的工作方式

延长每一相的通电时间，可以使电动机获得较多的能量，增加高频时输出的转矩。例如三相步进电动机的双三拍、六拍方式。

4.6.4 步进电动机的驱动方式

步进电动机使用脉冲电源驱动。脉冲电源中，开关管按照控制脉冲的规律“开”和“关”，使直流电源以脉冲方式向绕组供电。步进电动机的驱动方式有多种，可以根据实际需要选用。

1. 单电压驱动

单电压驱动电路最简单，工作时，只用一个电压电源对电动机绕组供电。

例如，在图 4.51 中，只有一个电源 V ；电路中的限流电阻 R_1 使绕组供电电流减小，使电动机的高频性能下降；在 R_1 两端并联电容，可以使电流的上升波形变陡，改善高频特性，但使低频性能变差。

限流电阻要损耗一定能量，电路效率低，一般只用于小功率步进电动机的驱动。

2. 双电压驱动

提高电压可以使绕组中的电流上升波形变陡，双电压驱动有两种方式：双电压法和高低压法。

(1) 双电压法

双电压法的基本思路是，在低频段使用较低的电压驱动，在高频段使用较高的电压驱动。其原理如图 4.52 所示。

当电动机工作在低频时，关断 V_1 ，这时电动机的绕组由低电压 V_L 供电，控制脉冲通过 V_2 使绕组得到低压脉冲电源。当电动机工作在高频时，打开 V_1 ，则二极管 VD_2 反向截止，切断低电压电源 V_L ，电动机绕组由高电压 V_H 供电，控制脉冲通过 V_2 使绕组得到高压脉冲电源。

应用双电压法，低频段仍然具有单电压驱动的特点，高频段则具有良好的高频性能，但在限流电阻 R 上仍然会产生损耗和发热。

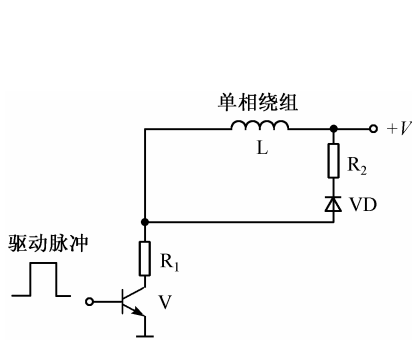


图 4.51 步进电动机驱动电路

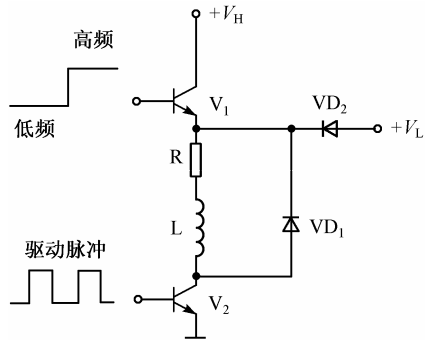


图 4.52 双电压驱动原理图

(2) 高低压法

高低压法的电路原理如图 4.53 所示，看起来与双电压法相似，但控制原理有很大差别。其基本思路是，不论电动机工作的频率高低，在绕组通电开始时都用高压供电，使绕组中电流迅速上升，然后用低压来维持绕组中的电流。

高压开关管 V_1 的输入脉冲 u_H 与低压开关管 V_2 的控制脉冲 u_L 同时起步，但 u_H 脉宽要窄得多。两个脉冲使开关管 V_1 、 V_2 同时导通，使高电压 V_H 为电动机绕组供电，绕组中电流 i 快速上升，电流波形的前沿很陡（如图 4.53 所示）。窄脉冲 u_H 结束时，高压开关管 V_1 截止，低

电压 V_L 通过二极管 VD_2 为绕组继续供电，只需数伏就可以为绕组提供较大电流。

通过计算平均电流来选择低电压 V_L 值，以防止低频时绕组通电周期长，而产生能量过剩；高频时则要保证在最高工作频率工作时， u_H 的脉宽不要大于 u_L 的脉宽。

步进电动机以脉冲方式供电，电源电压是其最高电压，而不是平均电压，所以，只要不偏离额定值太远，步进电动机都可以工作。一般选择范围高压约 80~150 V，低压约 5~20 V。

高低压驱动法目前应用比较普遍，缺点是低频时电流有较大的上冲，噪声较大，存在低频共振现象；在高压与低压交接处会引起输出转矩下降；设备成本也较高。

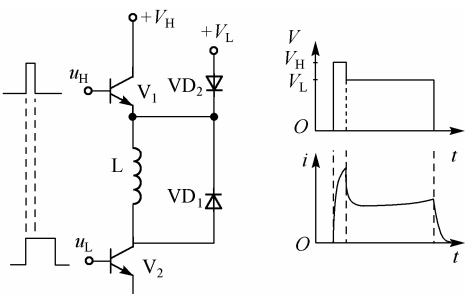


图 4.53 高低压驱动原理图

3. 斩波驱动

斩波恒流驱动的原理如图 4.54(a)所示。 V_1 是一个高频开关管，用于斩波；控制开关管 V_2 的发射极接小电阻 R 到地， R 也是电动机绕组电流的取样电阻；比较器的一端接给定电压 u_c ，另一端接取样电阻 R 上的压降，当这个取样电压为 0 时，比较器输出高电平。

当控制脉冲 u_i 为高电平时， V_1 和 V_2 两个开关管均导通，电源向绕组供电。由于绕组电感的作用，电阻 R 上的电压 u_R 逐渐升高，当 u_R 超过电压 u_c 的值时，比较器输出低电平， V_1 截止，电源被切断；当 R 上的电压降落到 u_R 小于电压 u_c 时，比较器输出高电平， V_1 又导通，电源又开始向绕组供电。这样反复循环，直到 u_i 为低电平，如图 4.54(b)所示。

电源脉冲式供电提高了电源效率，并且有效地抑制共振；无须外接影响时间常数的限流电阻，提高了高频性能。但绕组的电流波形为锯齿形，会产生较大的电磁噪声。

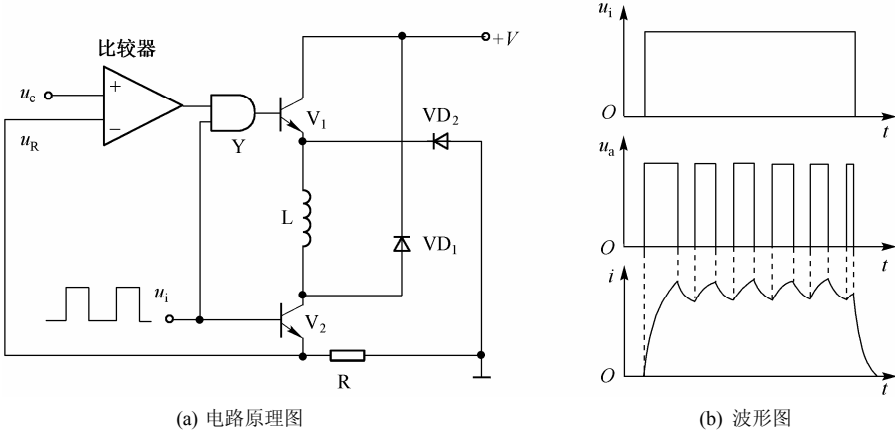


图 4.54 斩波恒流驱动原理图

4. 细分驱动

细分驱动是将一个步距角细分成若干小步的驱动方法，可以大大提高对执行机构的控制精度，也可以减小或消除振荡、噪声和转矩波动。目前，采用细分驱动技术已经可以将原步距角细分成数百小步。

如果在斩波恒流驱动电路中，用一个 N 个台阶的阶梯电压作为比较器的给定电压，则

绕组中将得到阶梯形电流。通电相的电流并不马上升到最大，而断电相的电流并不立即降为 0，此时它们产生的磁场合力，使转子在原来的步距角范围内到达一个新的平衡位置。电流每升或降一个阶级，转子转动一小步，按这样的规律转子转过 N 小步，才相当于它转过一个步距角。

例如，三相步进电动机转子有 40 个齿，六拍工作方式的步距角为 1.5° 。如果将步距角 4 细分，则一小步为 0.375° 。细分后各绕组的电流波形如图 4.55 所示。

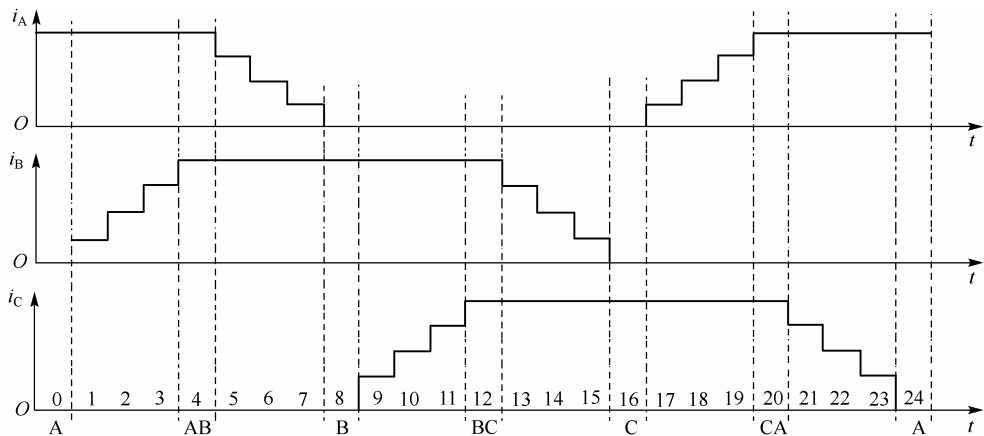


图 4.55 三相六拍 4 细分各绕组电流波形

4.6.5 步进电动机的控制

步进电动机的控制要素包括工作方式、旋转方向、位置和速度。控制输出脉冲信号的分配组合、顺序、次数和频率就可以方便地实现这些控制，大多数专用驱动芯片还将驱动和保护集于一体。

1. 工作方式控制

按照工作方式对步进电动机各相通电的过程称为脉冲分配。实现的方法有两种：软件法和硬件法。

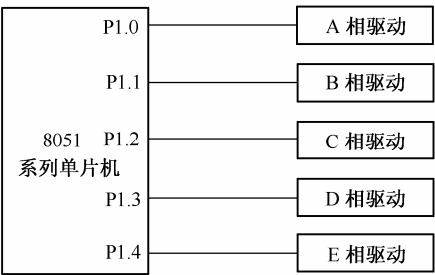


图 4.56 软件实现脉冲分配示意图

(1) 通过软件实现脉冲分配

编制程序，按照工作方式给定的通电换相顺序，通过计算机的 I/O 口向步进电动机各相驱动电路发出控制脉冲。

【例 4.6】利用 8051 系列单片机的 P1.0~P1.4，向五相步进电动机传送控制信号，如图4.56 所示。

五相十拍工作方式通电换相的正序为 AB—ABC—BC—BCD—CD—CDE—DE—DEA—EA—EAB，共 10 个通电状态。如果在 P1 口输出的控制

信号中，0 代表使绕组通电，1 代表使绕组断电，则可用 10 个控制字来对应这 10 个通电状态，如表 4.8 所示。

在程序中，依次将这 10 个控制字送到 P1 口，每完成一个循环，步进电动机就会转动一个齿距角，每送一个控制字，就完成一拍，步进电动机转过一个步距角。

表 4.8 五相十拍工作方式的控制字

通 电 状 态	P1.4(E)	P1.3(D)	P1.2(C)	P1.1(B)	P1.0(A)	控 制 字
AB	1	1	1	0	0	FCH
ABC	1	1	0	0	0	F8H
BC	1	1	0	0	1	F9H
BCD	1	0	0	0	1	F1H
CD	1	0	0	1	1	F3H
CDE	0	0	0	1	1	E3H
DE	0	0	1	1	1	E7H
DEA	0	0	1	1	0	E6H
EA	0	1	1	1	0	EEH
EAB	0	1	1	0	0	ECH

软件法在电动机运行过程中，要不停地产生控制脉冲，占用了大量的 CPU 时间，硬件法可以避免这种缺点。

(2) 通过硬件实现脉冲分配

硬件法就是使用脉冲分配器芯片进行通电换相控制。CPU 只需提供步进脉冲，进行速度控制和转向控制，脉冲分配的工作交给脉冲分配器来自动完成。

已经有多种脉冲分配器的集成电路芯片，例如，用于控制三相和四相步进电动机的单极性控制电路 8713 可以选择以下不同的工作方式。

- 三相步进电动机：单三拍、双三拍、六拍；
- 四相步进电动机：单四拍、双四拍、八拍。

8713 还可以选择单时钟输入或双时钟输入，具有正反转控制、初始化复位、工作方式和输入脉冲状态监视等功能。所有输入端内部都设有施密特整形电路，提高抗干扰能力，使用 4~18 V 直流电源，输出电流为 20 mA。

2. 转向控制

按给定的工作方式正序通电换相，步进电动机就正转；如果按相反次序通电换相，则反转。例如，四相步进电动机工作在单四拍方式，按正序 A—B—C—D，电动机就正转；反转则按反序 A—D—C—B。

3. 位置控制

步进电动机的一大优点，就是只需简单的开环控制就能达到足够的位置精度。当步进电动机带动执行机构从一个位置精确地运行到另一个位置时，需要控制的是：

- 绝对位置，即步进电动机控制的执行机构当前的位置参数。绝对位置的极限由执行机构运动的范围决定，超越极限就应报警；
- 位移量，即从当前位置移动到目标位置的距离，再折算成步进电动机的步数。折算参数可由键盘或电位器输入。

位移量控制的一般方法是，步进电动机每走一步，步数减 1。如果没有失步存在，则执行机构到达目标位置时，步数正好减到 0。因此，用步数等于 0 来判断是否移动到目标位，作为步进电动机停止运行的信号。

绝对位置可作为人机对话的显示参数，或作为其他控制目的的重要参数（例如，越界报警）。当步进电动机正转时，每走一步，绝对位置加 1；而当反转时，则每次步进减 1。

4. 速度控制

控制脉冲的频率决定了步进电动机的转速。脉冲的时间间隔越短，步进电动机就转得越快。

软件脉冲分配方式的调速，可以通过采用延时或定时程序调整两个控制字之间的时间间隔来实现；硬件脉冲分配方式的调速，可以通过控制步进脉冲的频率来实现。

步进电动机驱动执行机构从 A 点到 B 点移动时，要经历升速、恒速和减速过程。如果启动频率超过极限启动频率 f_q ，步进电动机会产生失步，不能正常启动；如果突然停止，会由于惯性发生过冲，造成位置精度降低；升降速过于缓慢，则会影响执行机构的工作效率。对步进电动机的加减速应在保证不失步和不过冲的前提下，尽快移动到指定位置。

为了满足加减速要求，步进电动机的运行通常按照根据经验和试验得到的加减速曲线进行。如图 4.57(a)所示的匀加减速曲线和图 4.57(b)所示的 S 形（分段指数曲线）加减速曲线。

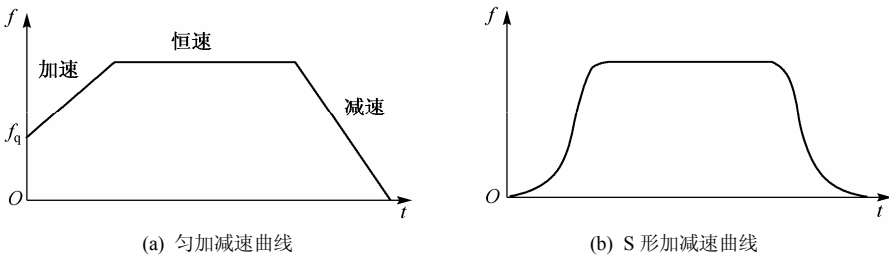


图 4.57 加减速运行曲线

因为电动机的电磁转矩与转速的关系接近指数规律，所以采用指数加减速曲线或 S 形（分段指数曲线）加减速曲线是最好的选择。通常用离散法将加、减速曲线离散化。将离散所得

的转速序列所对应的定时常数列，做成表格存储在程序存储器中。在程序运行中，使用查表的方式重装定时常数，这样比用计算法节省时间，有利于提高系统的响应速度。

近似指数加速曲线如图 4.58 所示，离散后速度并不是一直上升的，而是每升一级都要在该级上保持一段时间，因此实际加速轨迹是阶梯状曲线。如果速度是等间距分布，则在每个速度级上保持的时间并不等长。可以用该速度级数 N 与常数 C 的乘积去近似在该速度级保持的时间，让步进电动机在每个速度级上走 NC 步。

减速可以简化为加速的逆过程。

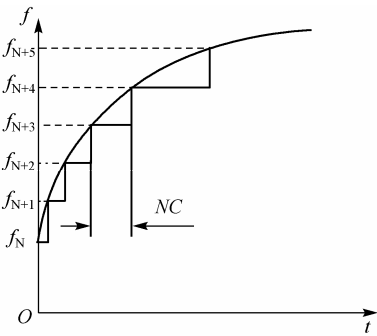


图 4.58 加速曲线离散化

4.7 无刷直流电动机控制

无刷直流电动机利用电子换向器取代了机械的电刷和换向器，避免了它们引起的致命弱点，既保留了直流电动机机械特性线性好、调速范围宽、启动转矩大、控制电路简单等优点，又具有交流电动机的结构简单、运行可靠、维护方便等优点，因而自 1962 年问世以来，就以极快的速度发展和普及到各个领域。

4.7.1 无刷直流电动机的原理与结构

1. 工作原理

为了驱动直流电动机产生恒定的转矩，必须使固定磁场和电枢磁场的方向始终保持相互

垂直。普通直流电动机的定子产生的磁场固定不动，通过换向器和电刷不断地改变电枢绕组中电流的方向来做到这一点。

无刷直流电动机的结构正好与普通直流电动机相反：转子是永磁体，电枢在定子上。通过定子电枢各相绕组不断地换相通电，使定子电枢磁场与转子永磁磁场始终保持 90° 左右的空间角，同样可以驱动转子旋转。

图 4.59 中，将三相无刷直流电动机的转子简化为只有一对磁极，定子绕组分别为 A 相、B 相、C 相。工作时，3 个在空间间隔 120° 的转子位置传感器 V1, V2, V3，通过驱动开关 VA, VB, VC 控制对应相绕组的通电与断电。

当转子处于图 4.59(a)所示的位置时，传感器 V1 输出高电平使开关 VA 导通，A 相绕组通电，而 B, C 两相处于断电状态。A 相绕组通电使定子产生的磁场与转子的永磁磁场相互作用，产生的转矩推动转子逆时针转动。

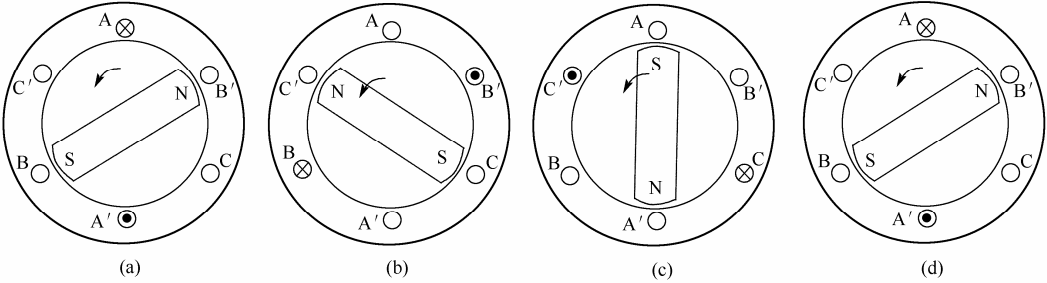


图 4.59 通电绕组与转子位置关系

当转子转到图 4.59(b)的位置时，传感器 V1 输出低电平使开关 VA 截止，A 相断电；同时，V2 输出高电平使开关 VB 导通，B 相通电；C 相仍然断电。由于通电相随着转子位置而变化，使定子磁场方向也发生了变化，与转子永磁磁场相互作用，以同样大的转矩，继续推动转子逆时针转动。

当转子转到图 4.59(c)的位置时，传感器 V1, V2 输出低电平，V3 输出高电平，因此，A 相、B 相断电，C 相通电，定子磁场方向又发生变化，继续推动转子转到图 4.59(d)的位置，使转子转过一周又回到原来位置。如此循环，驱动电动机转动。

图 4.60 给出的顺序换相通电过程中，定子各相绕组在工作气隙中所形成的旋转磁场是跳跃式运动的。旋转磁场在一周内 3 种状态，每种磁状态持续 120° ，即 $1/3$ 周。

2. 结构

无刷直流电动机结构框图如图 4.61 所示，直流电源通过开关电路向电动机定子绕组供电，位置传感器随时检测转子所处的位置，据此来控制开关管的导通和截止，从而实现定子相应绕组通电、断电的电子换向。

无刷直流电动机的基本结构如图 4.62 所示，其中，定子上开有齿槽，齿槽数应是转子极数和相数的整数倍，绕组的相数有三相、四相、二相及五相。转子的结构有两种：将瓦片状永磁体贴在外表的凸极式转子和永磁体嵌入铁心的内嵌式转子（如图 4.63 所示）。

3. 位置传感器

位置传感器是无刷直流电动机的关键部分，目前应用最多的是霍尔式位置传感器。

作为传感元件的开关型霍尔集成电路外形像一个普通晶体管[如图 4.64(a)所示]，结构简单，性能可靠，成本低。

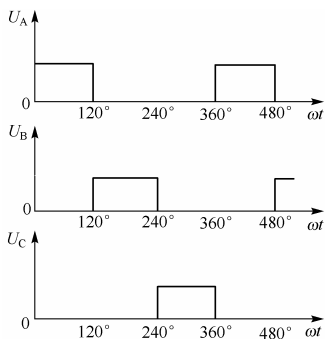


图 4.60 各相相电压波形

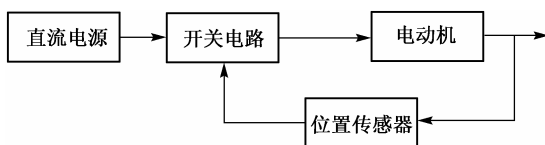


图 4.61 无刷直流电动机结构框图

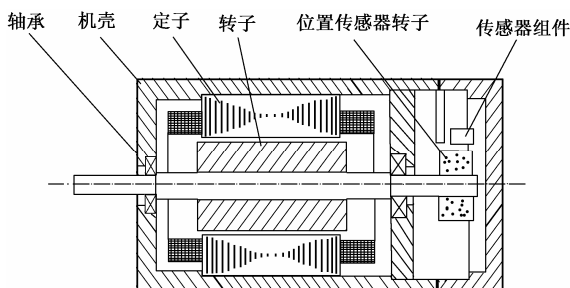


图 4.62 无刷直流电动机结构示意图

在长方形半导体薄片上通入电流 I , 当在垂直于薄片的方向上施加磁感应强度为 B 的磁场时, 则在与电流 I 和磁场强度 B 构成的平面相垂直的方向上会产生一个电动势 E , 这种效应称为霍尔效应。 E 称为霍尔电动势, 其大小为

$$E = K_H IB \quad (4.14)$$

式中, K_H 为灵敏度系数。

当磁场强度方向与半导体薄片不垂直, 而是成 θ 角时, 霍尔电动势的大小改为

$$E = K_H IB \cos \theta \quad (4.15)$$

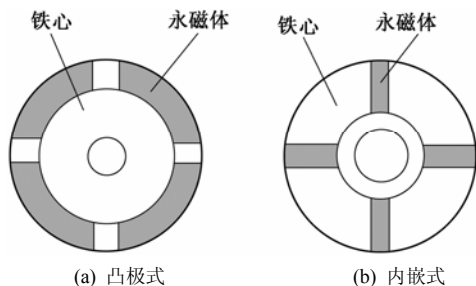


图 4.63 永磁转子结构类型

对可以感知永磁转子产生的磁场变化的霍尔半导体通入直流电, 当磁场强度大小和方向随着转子的位置不同而变化时, 霍尔电动势的大小和相位随之发生变化, 从而检测出转子位置, 其电路原理如图 4.64(b) 所示。

其他位置传感器还有利用光电效应的光电式位置传感器和利用电磁效应的电磁式位置传感器。

4.7.2 无刷直流电动机的绕组连接方式

无刷直流电动机有多种结构, 每种电动机可分为半桥驱动和全桥驱动, 全桥驱动又可分为星形和角形连接及不同的通电方式。

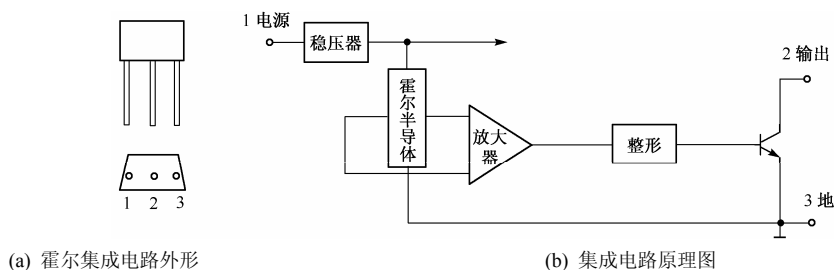


图 4.64 霍尔集成电路

半桥式驱动方式结构简单，但每相每转通电时间少，绕组利用率低，输出转矩波动较大。因此，全桥式驱动实际应用较多。

全桥式驱动下的绕组又分为星形连接和角形连接，仍以三相无刷直流电动机为例，下面分析各种连接方式的特点。

1. 三相星形连接全桥驱动方式

图4.65 是三相星形连接全桥驱动电路。图中，开关管 V_1, V_3, V_5 是 P 沟道 MOSFET 功率管，低电平时导通；开关管 V_2, V_4, V_6 是 N 沟道 MOSFET 功率管，高电平时导通。3 个位置传感器的输出控制这些开关管的导通和截止，其控制方式有两种：两两导通方式和三三导通方式。

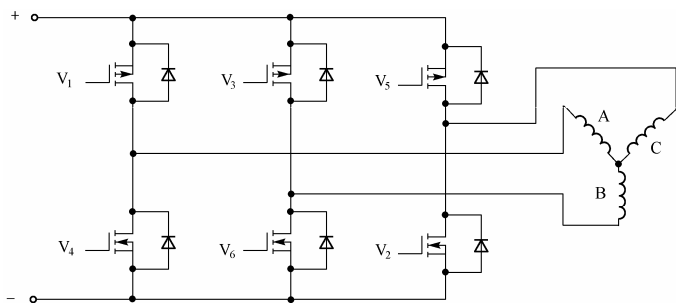


图 4.65 三相星形连接全桥驱动电路

(1) 两两导通方式

两两导通方式是每次使 2 个开关管同时导通，共有 6 种导通状态，导通顺序为 $V_1, V_2 \rightarrow V_2, V_3 \rightarrow V_3, V_4 \rightarrow V_4, V_5 \rightarrow V_5, V_6 \rightarrow V_6, V_1$ 。每隔 60° 改变一次导通状态，每改变一次状态更换一个开关臂，每个开关管导通 120° 。

当 V_1, V_2 导通时，电流的路线为电源 $\rightarrow V_1 \rightarrow A$ 相绕组 $\rightarrow C$ 相绕组 $\rightarrow V_2 \rightarrow$ 地。设各绕组产生的电磁转矩 $T_A = T_B = T_C$ ，且电流流入绕组时的电磁转矩为正，流出时电磁转矩为负，则合转矩 T_{12} 如图 4.66(a)所示，其大小为 $T_{12} = \sqrt{3}T_A$ 。

当转子转过 60° 角后，位置传感器送过来的控制信号使 V_1 截止， V_2, V_3 导通，电流的路线变为电源 $\rightarrow V_3 \rightarrow B$ 相绕组 $\rightarrow C$ 相绕组 $\rightarrow V_2 \rightarrow$ 地，则合成的电磁转矩 T_{23} 如图 4.66(b)所示，大小仍为 $T_{23} = \sqrt{3}T_A$ ，但方向逆时针转过 60° 。

导通状态每换一次，合成电磁转矩的矢量方向就转过 60° 角。6 个导通状态下合成电磁转矩的矢量方向如图 4.66(c)所示，但它们的大小都是 $\sqrt{3}T_A$ 。

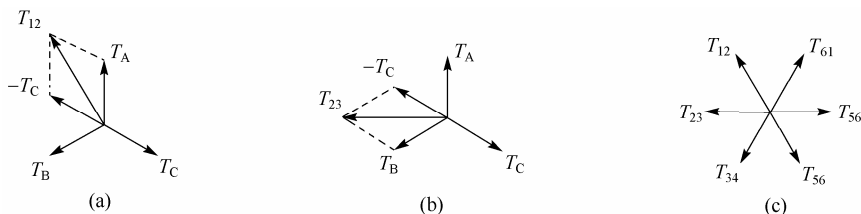


图 4.66 三相星形连接两两导通转矩矢量图

与三相半桥式驱动方式相比较，三相全桥星形连接两两导通方式下，每个开关管导通 120° ，每相绕组通电 240° ，增加了绕组的利用率。合成转矩大小增加为 $\sqrt{3}$ 倍，空间分布为间隔 60° ，转矩的波动大大减小，其输出转矩的波形如图 4.67 所示。

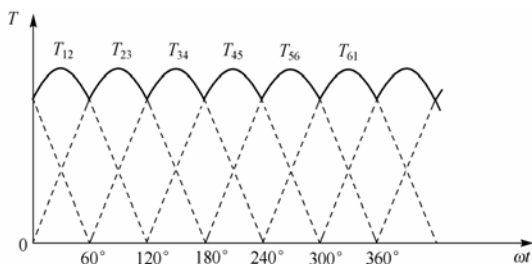


图 4.67 三相全桥式驱动的转矩波形

(2) 三三导通方式

三三导通方式指的是每次使 3 个开关管同时导通，也有 6 种导通状态，导通顺序为 $V_1, V_2, V_3 \rightarrow V_2, V_3, V_4 \rightarrow V_3, V_4, V_5 \rightarrow V_4, V_5, V_6 \rightarrow V_5, V_6, V_1 \rightarrow V_6, V_1, V_2$ 。每隔 60° 改变一次导通状态，每改变一次状态更换一个开关管，每个开关管导通 180° 。

当 V_1, V_2, V_3 导通时，电流的路线为电源 $\rightarrow V_1, V_3 \rightarrow A$ 相绕组和 B 相绕组并联 $\rightarrow C$ 相绕组 $\rightarrow V_2 \rightarrow$ 地，其合成电磁转矩 T_{123} 如图 4.68(a) 所示，大小为 $1.5T_C$ ，方向与 T_C 相反。

转子转过 60° 角后，位置传感器控制信号使 V_1, V_5, V_6 截止， V_2, V_3, V_4 导通。电流的路线为电源 $\rightarrow V_3 \rightarrow B$ 相绕组 $\rightarrow A$ 相绕组和 C 相绕组并联 $\rightarrow V_4, V_5 \rightarrow$ 地，其合成电磁转矩 T_{234} 如图 4.68(b) 所示，大小为 $1.5T_B$ ，方向与 T_B 相同。

通电状态每变换一次，转子转过 60° ，6 个导通状态下合成电磁转矩的矢量方向如图 4.68(c) 所示，但它们的大小相同都是单相转矩的 1.5 倍。

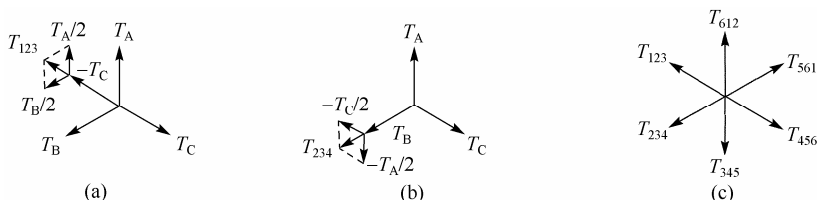


图 4.68 三相星形连接三三导通转矩矢量图

2. 三相角形连接全桥驱动方式

三相全桥驱动的另一连接方式是角形连接，如图 4.69 所示。角形连接的控制方式也有两两导通方式和三三导通方式。

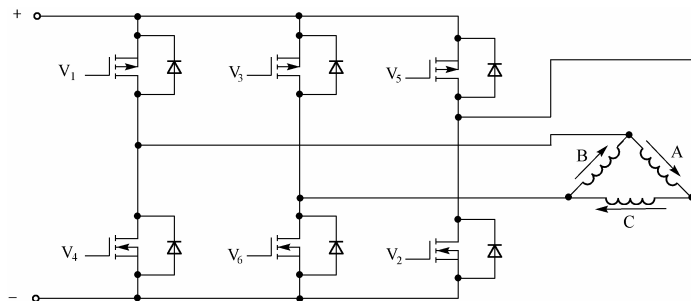


图 4.69 三相角形连接全桥驱动电路

(1) 两两导通方式

三相角形连接两两导通方式的各开关管导通顺序为 $V_1, V_2 \rightarrow V_2, V_3 \rightarrow V_3, V_4 \rightarrow V_4, V_5 \rightarrow V_5, V_6 \rightarrow V_6, V_1$ 。6 个合成电磁转矩矢量大小为 $1.5T_A$ ，方向如图 4.70 所示，可见与星形三三导通方式相似。

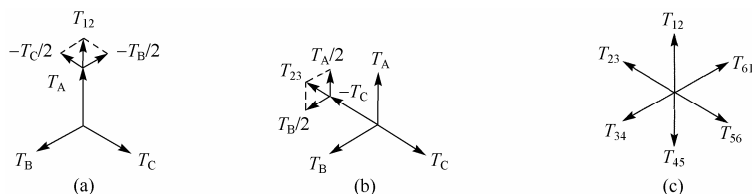


图 4.70 三相角形连接两两导通转矩矢量图

(2) 三三导通方式

三相角形连接三三导通方式的各开关管导通顺序为 $V_1, V_2, V_3 \rightarrow V_2, V_3, V_4 \rightarrow V_3, V_4, V_5 \rightarrow V_4, V_5, V_6 \rightarrow V_5, V_6, V_1 \rightarrow V_6, V_1, V_2$ 。6 个合成电磁转矩矢量大小为 $\sqrt{3}T_A$ ，方向如图 4.71 所示，与星形两两导通方式相似。

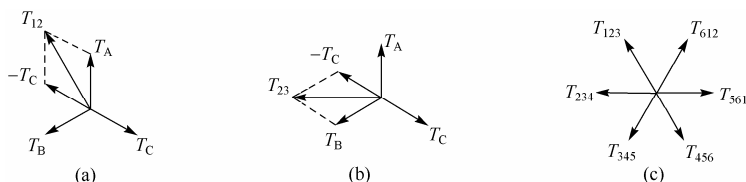


图 4.71 三相角形连接三三导通转矩矢量图

3. 多相绕组连接方式的选择

除了三相电动机外，还有四相和五相无刷直流电动机，它们也有多种连接方式。

设计应用系统时必须考虑选择不同多相绕组连接方式的性能和成本，目前实际应用较多的方式是三相星形全桥驱动。

(1) 绕组利用率

无刷直流电动机的绕组是断续通电的，适当地提高绕组通电利用率将使同时通电的导体数增加，电阻下降，效率提高。从这个角度来看，全桥比半桥好，相数越少绕组通电利用率越高。

(2) 转矩的波动

无刷直流电动机的输出转矩波动比普通直流电动机的大，因此，希望尽量减小转矩波动。

一般相数越多，转矩的波动越小，全桥驱动比半桥驱动转矩的波动小。

(3) 电路成本

相数越多，驱动电路所使用的开关管越多，成本越高；全桥驱动比半桥驱动所使用的开关管多一倍，成本要高；多相电动机的结构复杂，成本也高。

4.7.3 无刷直流电动机的控制

1. 单片机控制

选用带 PWM 口的单片机控制无刷直流电动机转速的实例如图 4.72 所示。

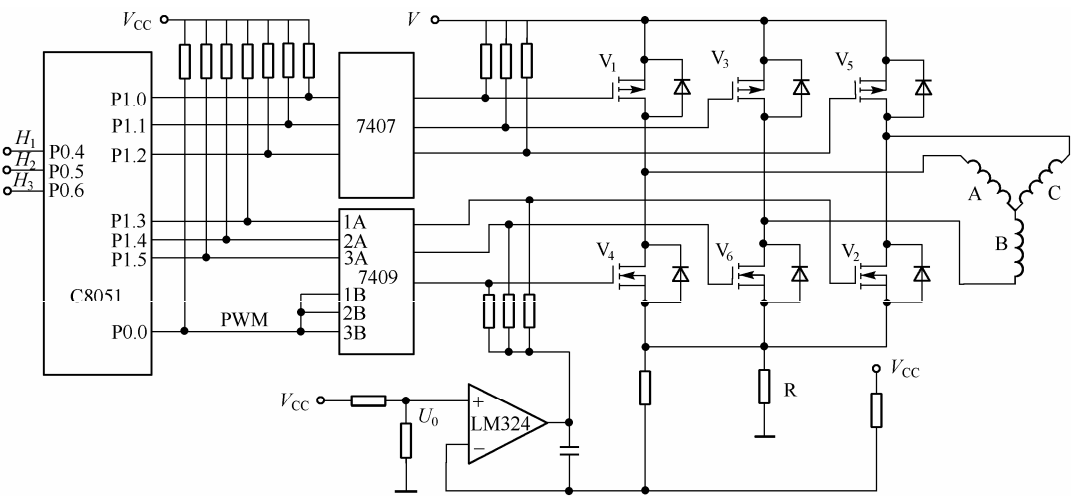


图 4.72 单片机控制无刷直流电动机原理图

C8051 的 P1.0~P1.2 作为输出口，通过驱动器 7407 控制全桥驱动电路上桥臂的 P 沟道 MOSFET (V_1, V_3, V_5)，输出 0 时 P 沟道 MOSFET 导通。P1.3~P1.5 通过与门 7409 控制下桥臂的 N 沟道 MOSFET (V_4, V_6, V_2)，输出 1 时 N 沟道 MOSFET 导通。P0.0 作为 PWM 输出口，控制电动机的转速。P0.4~P0.6 作为输入口，连接位置传感器输出的控制信号。C8051 的所有输出口都接上拉电阻，与 5 V 负载电平相匹配。控制电路所能实现的功能有：

(1) 换相控制

转子导通状态的转换和导通方式的切换都通过软件来完成：不断地根据位置传感器的输出信号 H_1, H_2, H_3 ，取相应的控制字送输出口 P1 来实现。例如，根据 P1 口与 MOSFET 管的连接关系，两两导通和二三导通方式的控制字分别如表 4.9、表 4.10 所示。

(2) 转速控制

和普通直流电动机一样，通过 PWM 方法来控制电枢的通电电流，可以实现无刷直流电动机的转速控制。

单片机的 PWM 口自动输出 PWM 波，电路中 MOSFET 管 V_4, V_6, V_2 的开关状态取决于单片机的控制字和 PWM 波与的结果。

(3) 转向控制

利用同一套位置传感器，不同的正反转对应逻辑关系分别编制正反转控制字表，改变开

关管的通电顺序就可以直接控制无刷直流电动机的正转和反转。

表 4.9 两两导通方式控制字

H ₁	H ₂	H ₃	导 通 管	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0	控 制 字
1	0	1	V ₁ , V ₂	0	0	1	1	1	0	0EH
1	0	0	V ₂ , V ₃	0	0	1	1	0	1	0DH
1	1	0	V ₃ , V ₄	1	0	0	1	0	1	25H
0	1	0	V ₄ , V ₅	1	0	0	0	1	1	23H
0	1	1	V ₅ , V ₆	0	1	0	0	1	1	13H
0	0	1	V ₆ , V ₁	0	1	0	1	1	0	16H

二二导通方式控制字（正转）

H ₁	H ₂	H ₃	导 通 管	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0	控 制 字
1	0	1	V ₄ , V ₅	1	0	0	0	1	1	23H
0	0	1	V ₃ , V ₄	1	0	0	1	0	1	25H
0	1	1	V ₂ , V ₃	0	0	1	1	0	1	0DH
0	1	0	V ₁ , V ₂	0	0	1	1	1	0	0EH
1	1	0	V ₆ , V ₁	0	1	0	1	1	0	16H
1	0	0	V ₅ , V ₆	0	1	0	0	1	1	13H

二二导通方式控制字（反转）

表 4.10 三三导通方式控制字

H ₁	H ₂	H ₃	导 通 管	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0	控 制 字
1	0	1	V ₁ , V ₂ , V ₃	0	0	1	1	0	0	0CH
1	0	0	V ₂ , V ₃ , V ₄	1	0	1	1	0	1	2DH
1	1	0	V ₃ , V ₄ , V ₅	1	0	0	0	0	1	21H
0	1	0	V ₄ , V ₅ , V ₆	1	1	0	0	1	1	33H
0	1	1	V ₅ , V ₆ , V ₁	0	1	0	0	1	0	12H
0	0	1	V ₆ , V ₁ , V ₂	0	1	1	1	1	0	1EH

三三导通方式控制字（正转）

H ₁	H ₂	H ₃	导 通 管	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0	控 制 字
1	0	1	V ₄ , V ₅ , V ₆	1	1	0	0	1	1	33H
0	0	1	V ₃ , V ₄ , V ₅	1	0	0	0	0	1	21H
0	1	1	V ₂ , V ₃ , V ₄	1	0	1	1	0	1	2DH
0	1	0	V ₁ , V ₂ , V ₃	0	0	1	1	0	0	0CH
1	1	0	V ₆ , V ₁ , V ₂	0	1	1	1	1	0	1EH
1	0	0	V ₅ , V ₆ , V ₁	0	1	0	0	1	0	12H

三三导通方式控制字（反转）

(4) 启动限流控制

与普通直流电动机一样，电动机启动时转速较低，转子磁通切割定子绕组所产生的反电

动势很小，可能产生过大的电流，必须加以控制。限流电路由采样电阻 R 和比较器组成。

当电动机启动时，电动机绕组的电流 I 在采样电阻 R 上的压降 IR 大于给定电压 U_0 ，比较器输出低电平，使 MOSFET 开关管 V_4, V_6, V_2 全部关断，电流迅速减小。当电流减小到 $IR < U_0$ 时，比较器恢复高电平输出。开关管恢复正常的通断顺序，从而电流被限制在 U_0/R 左右。

2. 专用集成电路

用于无刷直流电动机的专用集成电路芯片大多数是针对有霍尔式位置传感器的三相无刷直流电动机而设计的，多数具有换相、PWM 调速、转向控制、制动控制、电动机相数和工作方式选择、保护（如限流保护、欠压保护、过热保护）等功能。有些芯片还集成了驱动电路，可以方便地驱动小功率无刷直流电动机。

(1) 专用集成电路 LM621

LM621 专用于三相或四相无刷直流电动机的控制，对三相无刷直流电动机可以选择全桥或半桥驱动，角形或星形连接方式。对四相无刷直流电动机采用半桥驱动，其输出端提供 35 mA 基极电流，可以直接驱动双极型功率晶体管或功率 MOSFET 管，要求电动机使用霍尔式位置传感器，可以直接与外部 PWM 信号接口，实现调速功能，有转向控制功能、死区调节功能、过流保护功能和欠压保护功能，其原理框图如图 4.73 所示，与单片机连接驱动三相无刷直流电动机的电路如图 4.74 所示。

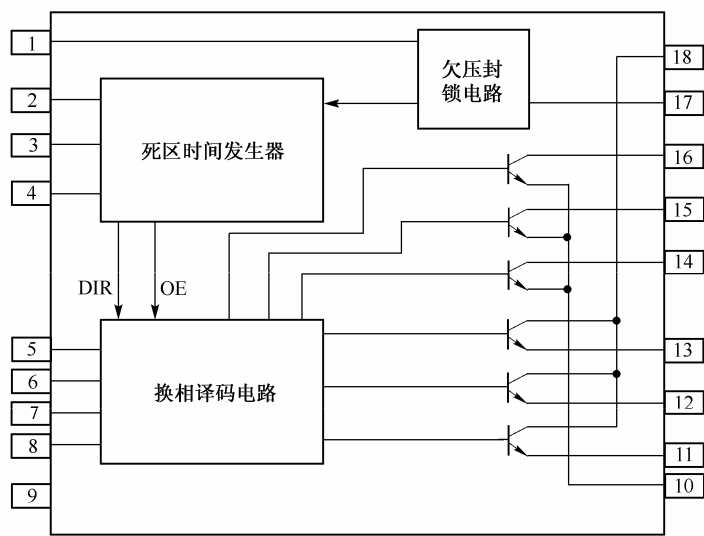


图 4.73 LM621 原理框图

(2) 无转子位置传感器专用集成电路

利用无刷直流电动机绕组断续通电，绕组线圈蓄能释放产生反电动势的一些特点，已经开发了各种无转子位置传感器的无刷直流电动机专用集成电路，有利于应用在体积受限制的领域。

例如，对于工作在三相全桥星形两两导通方式，三相星形连接有中点引出线的无刷直流电动机，专用集成电路 TDA5142T 可以利用电动机的反电动势信号控制换相，并产生精确测速频率信号，以用于闭环反馈控制。TDA5142T 内部设有启动电路，有制动、输出电流限制和热保护功能，可自动优化换相顺序，以适应不同电动机和不同负载的要求。

TDA5142T 是为控制大功率无转子位置传感器无刷直流电动机而设计的，必须外接功率 MOSFET 或功率晶体管。还必须借助于启动振荡器使电动机以步进电动机的方式启动，每一个起动脉冲都使电动机从一个通电状态转换成下一个通电状态，从而使电动机转动。当转动的电动机能产生足够大的反电动势时，电路自动切换到正常的无刷直流电动机工作状态。

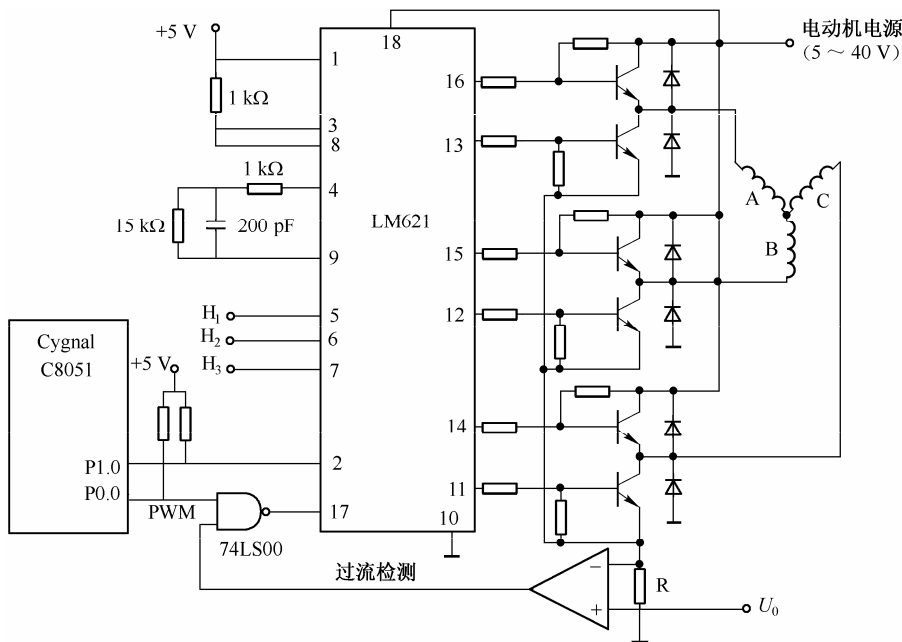


图 4.74 单片机控制 LM621 驱动三相无刷直流电动机

习题与思考题

- 1 全自动洗衣机的工作过程包括启动、进水、洗涤、排水和脱水等，试设计其工作流程。
- 2 试举出一个程序控制的应用实例，并分析其工作过程。
- 3 什么叫可编程控制器的响应时间？怎样满足系统对实时响应较高的要求？
- 4 如何通过 PWM 控制来实现直流电动机的调速？如何选择 PWM 的频率？
- 5 试比较双极性驱动方式、受限单极性驱动方式和受限倍频单极性驱动方式各有什么特点？它们各适用哪些场合？
- 6 交流电动机变频变压调速时，载波频率如何选择？调速后的机械特性有哪些不足？怎样改进？
- 7 简述交流异步电动机采用矢量控制的基本思路。
- 8 根据反应式步进电动机的工作原理，有哪些措施可以减小或消除振荡现象、提高负载能力、改善高频和低频性能、进行细分驱动。
- 9 为五相步进电动机设计三种不同的工作方式。这些不同的工作方式下，电动机的性能有何不同，为什么会产生不同？
- 10 四相无刷直流电动机有哪几种连接方式？试分析各种方式的特点。

第 5 章 常用控制技术

5.1 数字PID控制

5.1.1 PID控制

按偏差的比例、积分和微分进行控制，简称 PID 控制，其原理简单、易于实现、鲁棒性较强，在连续控制系统中技术最成熟，应用最广泛。随着微机控制技术的发展，将计算与逻辑控制功能结合的 PID 数字控制算法，已能用多种微控制器简单地实现。由于软件的灵活性，PID 算法还可以进行各种修正和改进，从而更加完善。

1. 优点

(1) 结构灵活

PID 控制的结构灵活，不仅可以用于常规的 PID 调节，而且可以根据系统的要求，采用各种 PID 的变型，如 PI 控制、PD 控制、不完全微分控制、积分分离式 PID 控制、带死区的 PID 控制、变速积分 PID 控制、比例 PID 控制，以及伪微分反馈控制 PDF（Pseudo-Derivative Feedback）和有前馈环节的 PDFF 等多种算法。

(2) 易于熟悉和掌握，整定方便

生产技术人员及操作人员都比较熟悉 PID 控制，并在实践中积累了丰富的经验，可以根据经验进行在线整定。

(3) 不要求出精确数学模型

到目前为止，仍有许多控制对象的精确数学模型难以建立。当系统参数发生变化时，进行模型辨识的代价又耗费很大，往往不能得到预期的效果，必须应用 PID 算法。

(4) 控制效果好

虽然计算机控制是离散的，但对于时间常数比较大的系统来说，近似于连续变化，用数字 PID 完全可以代替模拟调节器，并且更加灵活可靠。

2. 原理

PID 控制器是一种线性调节器，把设定值 x 与实际输出值 $y(t)$ 相减，得到控制偏差（或称误差） $e(t)$ 。 $e(t)$ 经比例、积分和微分后通过线性组合构成控制量 $u(t)$ ，对执行机构进行控制。

(1) P 调节（比例调节）

比例调节的控制规律为

$$u(t) = K_P e(t) + u_0 \quad (5.1)$$

式中， K_P 为比例增益； u_0 为控制常量，即误差为零时的控制变量。

比例调节对误差 $e(t)$ 是即时响应的。误差一旦产生，调节器立即产生控制作用，使被控制的变量 $y(t)$ 向减小误差的方向变化。只有当误差 $e(t)$ 发生变化时，控制变量 $u(t)$ 才变化。对于某些控制对象，存在静差。

静差也称为残余偏差或余差，是在控制动作结束后，仍然保留的一部分偏差。静差的概念可以用如图 5.1 所示的浮子杠杆式水位控制器为例来说明。

浮子 A 随水位变化而升降, 当水位过高时, 浮子 A 上升, 杠杆绕支点 B 逆时针转动, C 点下降, 通过阀杆使调节阀关小, 因而减小了进水量, 控制了水位。这里, 水位 $Y(t)$ 的波动是控制器的输入, 阀门开启程度的变化是控制器的输出 $u(t)$ 。水位与阀门开启程度的变化在数值成正比: 浮子升得越高, 阀门关闭得也越大。浮子的位移量和阀杆的位移量之间的比值取决于支点 B 的位置, 即杠杆的传动比, $\delta = a/b$ 称为比例度。

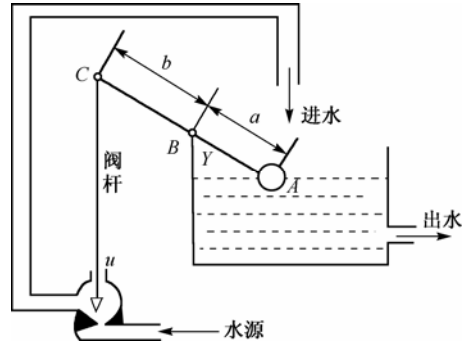


图 5.1 浮子杠杆式水位控制器

当出水流量增多时, 为了保持水位不下降过多, 控制器将自动把进水阀门开大一些。阀杆的上升是浮子下降的结果, 也是水位偏低的反映。也就是说, 出水负荷增大, 浮子杠杆控制动作结束后, 水位将低于给定值, 而不可能维持在原有的给定值; 反之, 出水负荷减小时, 水位将高于给定值。可见, 这种比例调节不能严格地把水位保持在原给定值, 也就是不能把误差消除为 0。

通常把比例度 δ 定义为偏差 $e(t)$ 与控制量 $u(t)$ 之比 [$e(t)$ 和 $u(t)$ 均指对于初始状态的变化量]。比例增益 $K_p = (1/\delta) \times 100\%$ 。

比例控制器的优点是反应快、结构简单, 缺点是不能完全消除静差。加大比例增益 K_p 可以减小静差, 但 K_p 过大, 会使系统动态特性变差, 导致系统不稳定。

(2) PI 调节 (比例积分调节)

为了消除比例调节器中残存的静差, 在比例调节的基础上加入积分调节, 比例积分 (PI) 调节的控制规律为

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(t) dt \right] + u_0 \quad (5.2)$$

式中, T_i 为积分常数, T_i 越大, 积分作用越弱。

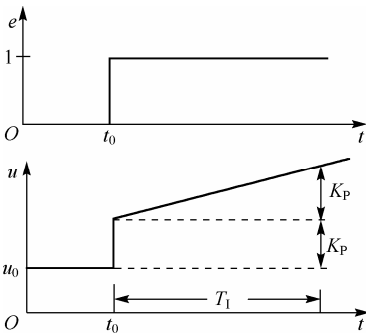


图 5.2 PI 调节的阶跃响应

PI 调节对误差的阶跃响应如图 5.2 所示。积分器的输出值取决于对误差的累积结果, 虽然误差不变, 但积分器的输出还在增大, 直至误差 $e = 0$ 。因此, 积分环节的加入能积累控制作用, 最终可消除静差, 提高系统的抗干扰能力, 但积分作用太强会使系统超调加大, 甚至出现振荡。

(3) PID 调节 (比例积分微分调节)

积分器虽然能消除静差, 但使系统的响应速度变慢。进一步的改进方法是通过检测误差的变化率来预报误差, 并对误差的变化做出响应, 即在 PI 调节的基础上再加上微分环节, 组成比例、积分、微分调节, PID 控制规律为

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_D \frac{de(t)}{dt} \right] + u_0 \quad (5.3)$$

式中, T_D 为微分常数, T_D 越大, 微分作用越强。

PD 调节器对误差阶跃响应如图 5.3 所示, 在误差 e 阶跃变化的瞬间, 有瞬时的冲激响应, 这就是微分环节调节器对误差的变化产生的控制作用 u_d 。误差变化越快, u_d 越大, 则反馈校

正量越大。加入微分环节有利于减小超调，克服振荡，从而改善了系统的动态性能；缺点是抗干扰能力变差。

PID 调节器也可以传递函数的形式表示为（设 $u_0=0$ ，如图5.4 所示）：

$$D(s)=\frac{U(s)}{E(s)}=K_P+\frac{K_P}{T_I}\frac{1}{s}+K_PT_Ds \tag{5.4}$$

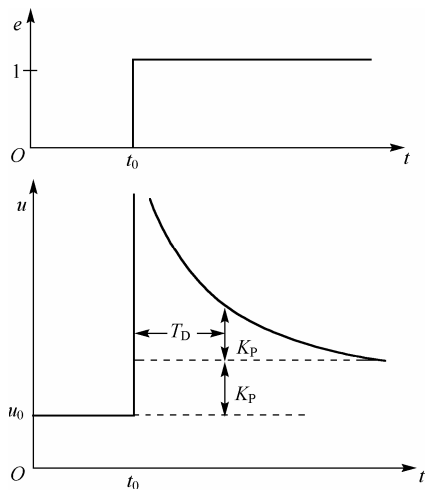


图 5.3 PD 调节器阶跃响应

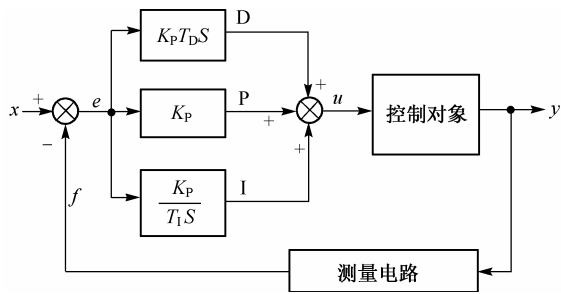


图 5.4 PID 控制的传递函数

5.1.2 PID算法的计算机实现

1. 数字 PID 位置型控制算法

为了在微机控制系统实现 PID 算法，将式(5.3)变换为差分方程

$$u(k)=K_P[e(k)+\frac{T}{T_I}\sum_{i=0}^k e(i)+T_D\frac{e(k)-e(k-1)}{T}] \tag{5.5}$$

式中， T 为采样周期， k 为采样序号。

作如下近似：

$$\int_0^t e(t)dt \approx \sum_{i=0}^k Te(iT)$$

$$\frac{de(t)}{dt} \approx \frac{e(iT)-e(iT-T)}{T}$$

式(5.5)称为数字 PID 位置式控制算法，如果控制系统的执行机构采用调节阀，则控制量对应阀门的开度，表征了执行机构的位置。

2. 数字 PID 增量型控制算法

式(5.5)需要累加偏差 $e(i)$ ，内存和计算时间花费较多，对之加以改进可得

$$\begin{aligned} \Delta u(k) &= u(k)-u(k-1) \\ &= K_P[e(k)-e(k-1)]+K_Ie(k)+K_D[e(k)-2e(k-1)+e(k-2)] \end{aligned} \tag{5.6}$$

式中，积分系数 $K_I=K_P\frac{T}{T_I}$ ，微分系数 $K_D=K_P\frac{T_D}{T}$ 。

式(5.6)称为数字 PID 增量型控制算法, 控制器每个采样周期输出的控制量, 是相对于上次控制量的增加, 可直接采用步进电机作为执行机构。

(1) 增量型算法的优点

- ① 增量的确定仅与最近几次误差采样值有关, 不会产生大的累加误差。计算误差或计算精度问题, 对控制量的计算影响较小。
- ② 得出的是控制量的增量, 误动作影响小, 必要时可以通过逻辑判断限制或禁止本次输出, 不会严重影响系统的工作, 而位置型算法的输出是控制量的全量输出, 误动作影响大。
- ③ 容易实现手动到自动的无冲击切换。

(2) 增量型算法的编程

整理式(5.6), 得差分方程

$$\Delta u(k) = q_0 e(k) + q_1 e(k-1) + q_2 e(k-2)$$

(5.7)

其中

$$q_0 = K_p \left(1 + \frac{T}{T_i} + \frac{T_D}{T} \right)$$
$$q_1 = -K_p \left(1 + \frac{2T_D}{T} \right)$$
$$q_2 = K_p \frac{T_D}{T}$$

编程序流程图如图5.5所示。

5.1.3 PID算法的改进

如果单纯地用微机控制系统去模仿模拟调节器, 一般说来, 控制质量不如采用模拟调节器, 这是因为

- 模拟调节器进行的控制是连续的。控制作用每时每刻都在进行, 而微机控制系统的控制量在一个采样周期内不变。数值的运算和输入/输出都需要一定的时间, 控制作用在时间上有延迟。
- 计算机的运算字长、A/D 转换、D/A 转换的分辨率和精度都有限制, 因而控制有误差。

只有发挥计算机运算速度快、逻辑判断功能强、编程序灵活等优势, 建立模拟调节器难以实现的改进控制规律, 才能使微机控制系统实现更好的性能。

1. 积分作用的改进

(1) 积分饱和

积分项的存在可能引起积分饱和。系统输出达到设定值后, 虽然偏差已经等于 0, 但由于积分项积累太大, 使计算得到的控制量仍保持较大的数值, 从而产生较大的超调, 延长了系统的调整时间。

产生积分项积累太大的原因如下。

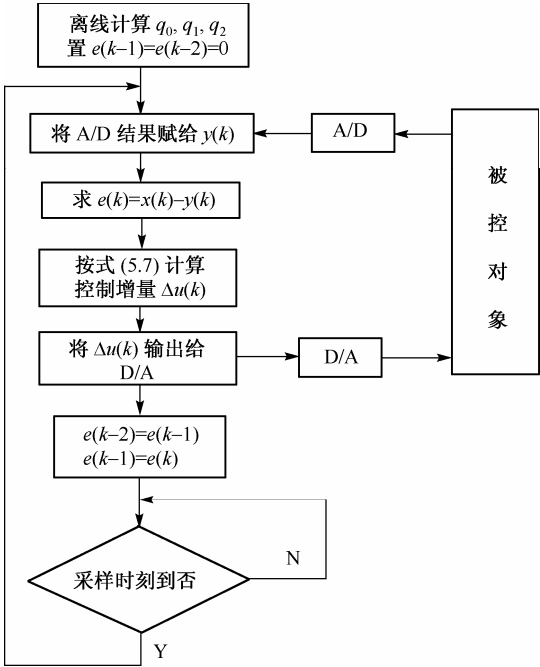


图 5.5 数字 PID 增量型控制算法程序流程图

① 一般的 PID 控制，在刚启动、较大的扰动或大幅度改变给定值时，偏差较大；而对于温度、成分等变化缓慢的过程，系统有较大的惯性和滞后，积分项积累时间长。

② 执行机构都存在极限，对于较大的偏差，即使 PID 运算得出的控制量 $u(k)$ 很大，实际控制作用也不可能超过极限，因而与控制作用完全实现时相比，将在较长的时间内保持较大误差值，积分项的累积也大大增加。

(2) 消除积分饱和的常用方法

① 积分分离法。积分分离即当偏差 $e(k)$ 较大时，取消积分；当偏差 $e(k)$ 较小时，才将积分作用投入，则

- 当 $|e(k)| > \beta$ 时，采用 PD 控制；
- 当 $|e(k)| < \beta$ 时，采用 PID 控制。

应根据具体对象及控制要求确定积分分离阈值 β 。若 β 值过大，则达不到积分分离的目的；若 β 值过小，则一旦被控量 $y(t)$ 无法跳出各积分分离区，只进行 PD 控制，将会出现残差。编写程序时，应当将积分项从数字 PID 差分方程式中分离出来，进行特殊处理。

② 遇限制削弱积分法。当计算控制量超过极限进入饱和区后，只执行削弱积分项的累加，而不进行增加积分项的累加。在计算 $u(k)$ 时，先判断 $u(k-1)$ 是否超过执行机构的极限 u_{\max} 或 u_{\min} ，若已超过 u_{\max} ，则只累计负偏差；若小于 u_{\min} ，则只累计正偏差，其算法框图如图 5.6 所示。这种方法可缩短系统处于饱和区的时间。

(3) 提高积分项的运算精度

提高运算精度可以充分发挥积分项减小残差的作用，方法如下：

① 原理上，将矩形积分改为梯形积分，其计算公式为

$$\int_0^t e(t)dt \approx \sum_{i=0}^k \frac{e(i) + e(i-1)}{2} \cdot T \tag{5.8}$$

② 计算上，

- 增加 A/D 转换位数，加长运算字长，以提高运算精度；
- 当积分项连续 n 次小于输出精度时，不作为“零”舍去，而是逐次累加，直到累加值大于输出精度才输出。

2. 微分作用的改进

标准 PID 控制算式的缺点是，对具有高频扰动的过程，微分作用响应过于灵敏，容易引起控制过程振荡，降低调节品质；计算机对每个控制回路输出时间是短暂的，而驱动执行器动作又需要一定时间，如果输出过大，则在短暂时间内执行器达不到应有指标，会使输出失真。

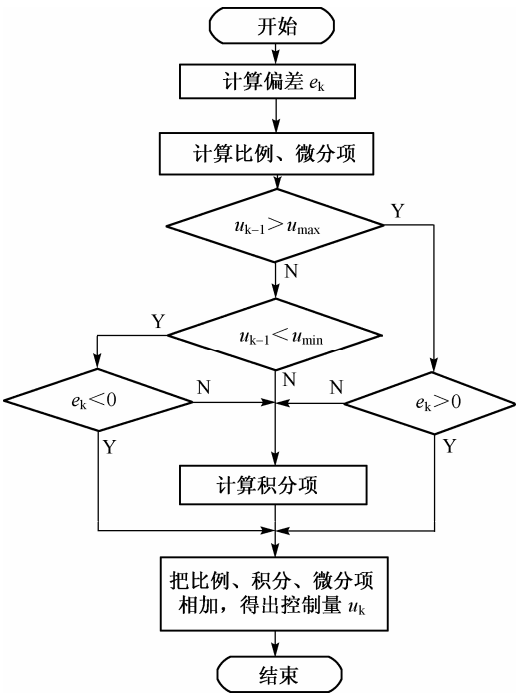


图 5.6 遇限制削弱积分法

(1) 不完全微分 PID 控制算法

为了克服这一缺点，可以在 PID 控制器中串接一阶惯性环节（低通滤波器），使微分作用来得较小而去得较慢。如图5.7所示，在 PID 控制器的微分项串接低通滤波器，构成的不完全微分项输出为

$$U(s) = \frac{T_D s}{1 + T_f s} E(s)$$

或

$$u(t) + T_f \frac{du(t)}{dt} = T_D \frac{de(t)}{dt} \quad (5.9)$$

对式(5.9)离散化，可得

$$u(kT) = \frac{T_f}{T + T_f} u(kT - T) + \frac{T_D}{T + T_f} [e(kT) - e(kT - T)] \quad (5.10)$$

当 $k \geq 0$ 时， $e(kT) = a$ ，由式(5.10)可得

$$u(0) = \frac{T_D}{T + T_f} a$$

$$u(T) = \frac{T_f T_D}{(T + T_f)^2} a$$

$$u(2T) = \frac{T_f^2 T_D}{(T + T_f)^3} a$$

⋮

显然， $u(kT) \neq 0, k = 1, 2, \dots$ ，并且

$$u(0) = \frac{T_D}{T + T_f} a \square \frac{T_D}{T} a$$

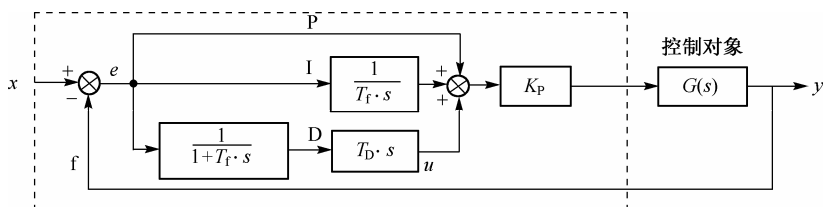


图 5.7 在微分项串接低通滤波器构成不完全微分

不同微分作用的 PID 控制阶跃响应如图5.8所示。

图5.8(a)和图5.8(b)分别表示标准 PID 控制算法和不完全微分 PID 控制算法在单位阶跃输入时的输出控制作用。由图5.8(a)可见，标准 PID 控制算法中的微分作用只在第一个采样周期内起作用，而且作用很强；而不完全微分 PID 控制算法的微分作用如图5.8(b)所示，在第一个采样周期内幅度小得多，但在较长时间内仍有作用，因此可获得较理想的控制效果。

(2) 微分先行 PID 控制算法

给定值的升降会给控制系统带来冲击，使超调量过大，执行机构动作剧烈。为了避免这些问题，对给定值变化频繁的系统可以采用微分先行 PID 控制算法。与标准 PID 控制的不同

之处在于，微分先行只对被控量 $y(t)$ 微分，而不对偏差 $e(t)$ 微分，也就是不对给定值 $x(t)$ 进行微分（如图5.9所示）。

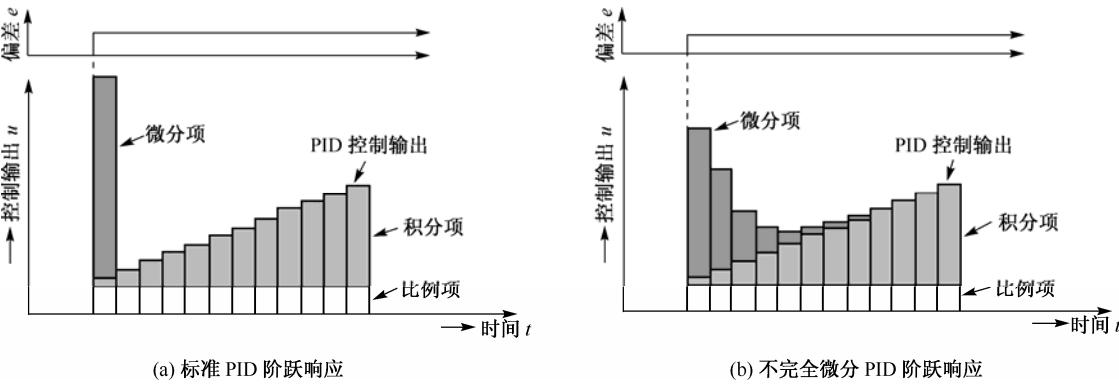


图 5.8 不同微分作用的 PID 控制阶跃响应

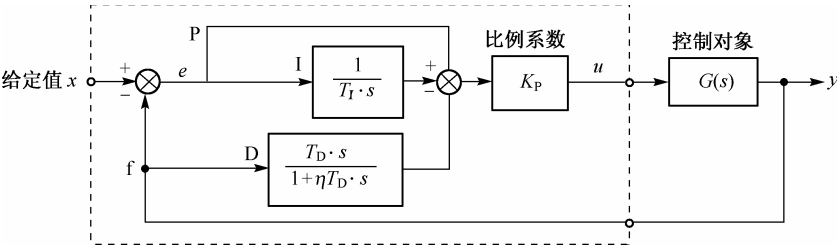


图 5.9 只对被控量微分的微分先行

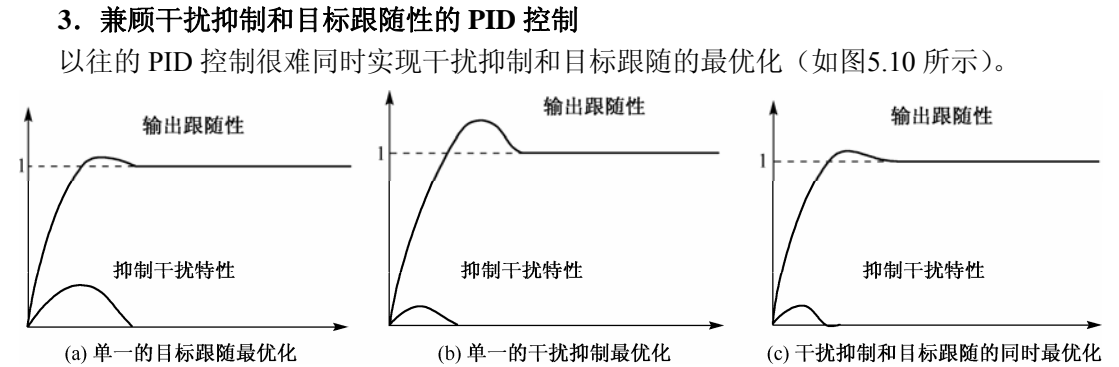


图 5.10 干扰抑制和目标跟随性的单一最优化和同时最优化

实现抗干扰性与跟随性分别独立调节的一种思路是，在 PID 控制器前加入目标值低通滤波器 $H(s)$ ，先调整 PID 控制器的参数使抗干扰性达到最优化，然后再调整目标值滤波器的参数实现跟随性的最优化，其框图和基本结构分别如图5.11和图5.12所示。

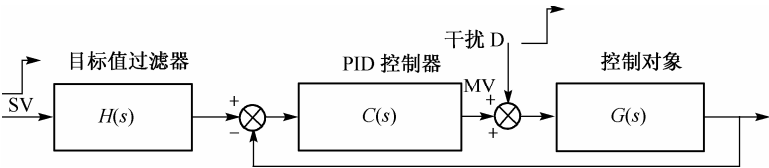


图 5.11 干扰抑制和目标跟随性分别独立调节的目标值过滤型 PID 控制框图

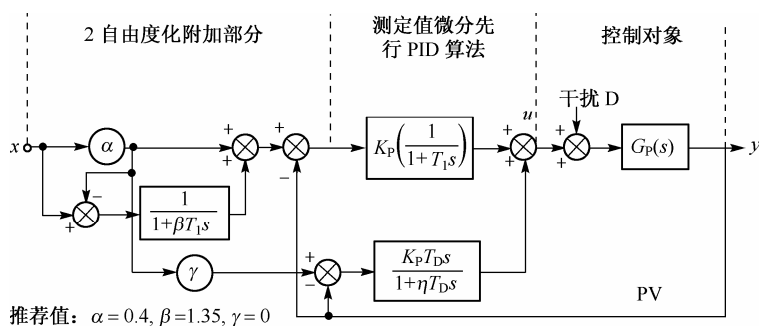


图 5.12 干扰抑制和目标跟随性分别独立调节的 PID 控制结构图

如图5.12所示, 输入给定值 x 与输出 y 间的传递函数 $Z_{sv}(s)$, 干扰 D 与输出 y 间的传递函数 $Z_D(s)$ 分别为

$$Z_{sv}(s) = \frac{K_p \left(\alpha + \left\{ \frac{1}{T_1 s} - \frac{(1-\alpha)(\beta-1)}{1+\beta T_1 s} \right\} + \frac{\alpha \gamma T_D s}{1+\eta T_D s} \right) G_p(s)}{1 + K_p \left(1 + \frac{1}{T_1 s} + \frac{T_D s}{1+\eta T_D s} \right) G_p(s)} \quad (5.11)$$

$$Z_D(s) = \frac{G_p(s)}{1 + K_p \left(1 + \frac{1}{T_1 s} + \frac{T_D s}{1+\eta T_D s} \right) G_p(s)} \quad (5.12)$$

式中, 微分系数 η 通常取值范围是 $0.1 \sim 0.125$ 。

目标值低通滤波器 $H(s)$ 中, 比例增益系数 α 、积分时间系数 β 、微分时间系数 γ 的取值说明如下:

- 当 $\alpha=1, \beta=1, \gamma=1$ 时, 就是基本的 PID 控制;
- 当 $\alpha=1, \beta=1, \gamma=0$ 时, 只对被控量微分的微分先行 PID 控制;
- 当 $\alpha=0, \beta=1, \gamma=0$ 时, 加入被控量比例的微分先行 PID 控制, 简化后其结构如图5.13所示;
- 当 $\alpha \neq 0, \beta=1, \gamma=0$ 时, P 控制实现抗干扰性与跟随性的独立;
- 当 $\alpha \neq 0, \beta=1, \gamma \neq 0$ 时, PD 控制实现抗干扰性与跟随性的独立;
- 当 $\alpha \neq 0, \beta \neq 1, \gamma=0$ 时, PI 控制实现抗干扰性与跟随性的独立;
- 当 $\alpha \neq 0, \beta \neq 1, \gamma \neq 0$ 时, PID 控制实现抗干扰性与跟随性的独立。

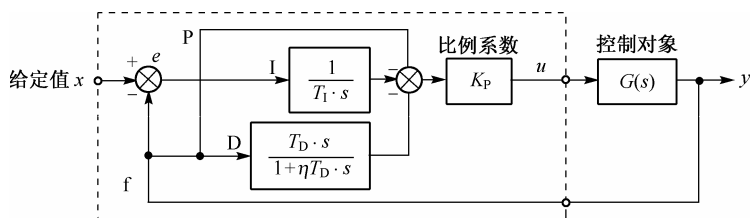


图 5.13 加入被控量比例的微分先行 PID 控制结构

可以看出, 系数 α, β, γ 的改变只会影响传递函数 $Z_{sv}(s)$, 而对外界干扰的传递函数 $Z_D(s)$

完全没有影响。因此，可以先调整 PID 参数优化抗干扰性，再通过调整系数 α, β, γ 优化跟随性，实现抗干扰性与跟随性分别独立的 PID 控制。

通常在开始调整时，选择 $\alpha=0.4, \beta=1.35, \gamma=0$ ，以防止过冲；而对于温度控制程序，常选择 $\alpha=0.4, \beta=1.35, \gamma=1.25$ 为起始值，然后微调达到跟随性最优。

抗干扰性最优、跟随性最优与兼顾干扰抑制和目标跟随性的 PID 控制响应的比较如图 5.14 所示。

如果以 $ITAE = \int t|e|dt$ 为控制特性评价指标画出等高线图，由图 5.15 可见，只有同时调节系数 α, β ，才可能到达控制特性的单峰极值点。

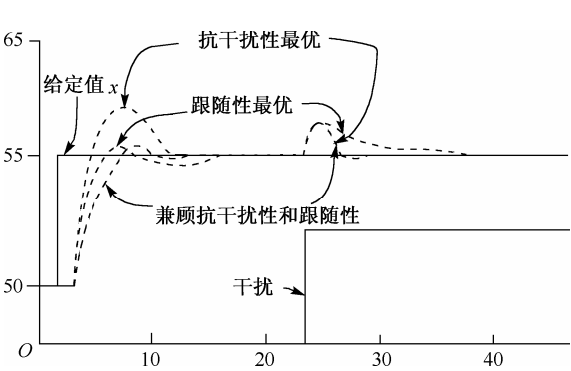


图 5.14 PID 控制响应比较

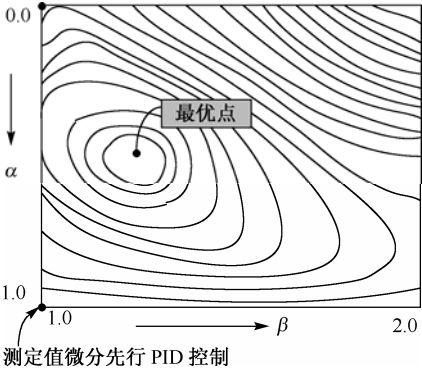


图 5.15 控制特性评价指标等高线图

4. 其他 PID 控制

(1) Bang-Bang 控制结合 PID 控制

这种算法使控制变量在偏差较大时取极限值，偏差小于一定范围时由 PID 决定，因而调节时间短，也可以保证足够的精度，这在给定值升降时特别有效。表达式为

$$|e(k)| = |r(k) - y(k)| \begin{cases} > a, & \text{Bang-Bang 控制} \\ \leq a, & \text{PID 控制} \end{cases} \tag{5.13}$$

(2) 带死区的 PID 控制算法

为了避免控制动作过于频繁，并消除由此引起的振荡，在微机控制系统中，可以采用带死区的 PID 算法：根据实验确定一个死区 ε ，当偏差绝对值 $|e(k)| \leq \varepsilon$ 时，令 PID 算式的输入误差为 0；当 $|e(k)| > \varepsilon$ 时，才将 $e(k)$ 输入 PID 算式。

ε 的具体数值可根据实际控制对象由实验确定：取得太小，调节过于频繁，达不到稳定被调节对象的目的；取得太大，系统将产生很大的滞后； $\varepsilon=0$ ，就是常规的 PID 控制。

5.1.4 PID 控制的参数整定

1. 采样频率的选择

(1) 采样周期的上限和下限

参见 2.2.1 节采样周期的选择。

(2) 其他因素

① 给定值的变化频率。加到被控对象上的给定值变化频率越高，采样频率应越高，以便迅速反映给定值的改变。

② 被控对象的特性。被控对象变化速度慢，采样周期一般取得较大。一些常用被控对象采样周期的经验数据如第 2 章表 2.1 所示。采样周期的选取还与系统纯滞后时间 τ 有关，通常 $T=(0.1\sim1)\tau$ ， τ 在系统中起作用的时间越长， T 取得就越长。

③ 执行机构的类型。执行机构动作惯性大，采样周期也应大一些，保证执行机构能反映控制输出值的变化。

④ 控制算法的类型。PID 算法的积分和微分作用与采样周期 T 的选择有关。选择采样周期 T 太小，由于计算精度的限制，偏差 $e(k)$ 可能始终为零，从而使微分积分作用不明显。还应考虑各种控制算法需要的计算时间。

⑤ 控制的回路数。采样周期应不小于所有控制回路执行控制程序的时间和输入输出时间之和。

2. PID 参数的工程整定

PID 参数的工程整定简单易行，不必依赖被控对象的数学模型。

(1) 扩充临界比例度法

① 选择采样周期为被控对象纯滞后时间的十分之一以下。

② PID 控制只保留比例作用，以选定的采样周期使系统工作。然后，逐渐增大比例增益 K_P ，直到系统发生持续等幅振荡。记下使系统发生振荡的临界比例增益 K_s 和临界振荡周期 T_s 。

③ 选择控制度。

控制效果的评价函数通常用误差平方积分 $\int_0^\infty e^2(t)dt$ 表示。

$$\text{控制度} = \frac{\left[\int_0^\infty e^2(t)dt \right]_{\text{DDC}}}{\left[\int_0^\infty e^2(t)dt \right]_{\text{模拟}}}$$

(5.14)

实际上，只用控制度表示控制效果的物理概念。例如，当控制度为 1.05 时，数字控制与模拟控制效果相当；当控制度为 2.0 时，数字控制比模拟控制效果差。

④ 根据选定的控制度，由表 5.1 求得 T ， K_P ， T_I ， T_D 的值。

表 5.1 按扩充临界比例度法整定参数

控 制 度	控 制 规 律	T	K_P	T_I	T_D
1.05	PI	$0.03T_s$	$0.53K_s$	$0.88T_s$	
	PID	$0.014T_s$	$0.63K_s$	$0.49T_s$	$0.14T_s$
1.2	PI	$0.05T_s$	$0.49K_s$	$0.91T_s$	
	PID	$0.043T_s$	$0.47K_s$	$0.47T_s$	$0.16T_s$
1.5	PI	$0.14T_s$	$0.42K_s$	$0.99T_s$	
	PID	$0.09T_s$	$0.34K_s$	$0.43T_s$	$0.20T_s$
2.0	PI	$0.22T_s$	$0.36K_s$	$1.05T_s$	
	PID	$0.16T_s$	$0.27K_s$	$0.40T_s$	$0.22T_s$

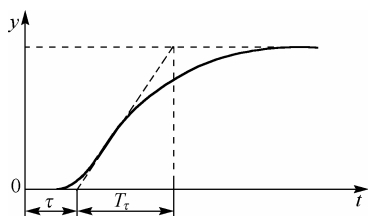


图 5.16 被调量在阶跃输入下的变化过程曲线

(2) 扩充响应曲线法

① 控制系统处于手动操作状态之下，将被调量稳定在给定值附近，然后突然改变给定值，给对象一个阶跃输入信号。

② 记录被调量在阶跃输入下的整个变化过程曲线，如图 5.16 所示。

③ 在曲线最大斜率处画切线，求得滞后时间 τ 、被控对象时间常数 T_τ 及它们的比值 T_τ / τ ，查表 5.2，即可得数字控制器的 K_P 、 T_I 、 T_D 及采样周期 T 。

表 5.2 按扩充响应曲线法整定参数

控制度	控制规律	T	K_P	T_I	T_D
1.05	PI	0.1τ	$0.84 T_\tau / \tau$	0.34τ	
	PID	0.05τ	$1.15 T_\tau / \tau$	2.0τ	0.45τ
1.2	PI	0.2τ	$0.78 T_\tau / \tau$	3.6τ	
	PID	0.16τ	$1.0 T_\tau / \tau$	1.9τ	0.55τ
1.5	PI	0.5τ	$0.68 T_\tau / \tau$	3.9τ	
	PID	0.34τ	$0.85 T_\tau / \tau$	1.62τ	0.65τ
2.0	PI	0.8τ	$0.57 T_\tau / \tau$	4.2τ	
	PID	0.6τ	$0.6 T_\tau / \tau$	1.5τ	0.82τ

(3) 归一参数整定法

归一参数整定法只需整定一个参数，是扩充临界比例度整定法的一种简化。如令 $T = 0.1 T_k$ ， $T_I = 0.5 T_k$ ， $T_D = 0.125 T_k$ ，式中 T_k 为纯比例作用下的临界振荡周期，则

$$\Delta u(k) = K_P [2.45e(k) - 3.5e(k-1) + 1.25e(k-2)]$$

改变参数 K_P ，观察控制效果，直到满意为止。该法可用于实现简易的自整定控制。

(4) 经验法

PID 各参数 T 、 K_P 、 T_I 、 T_D 对系统有不同的影响。

增大比例系数 K_P 有利于减小静差，一般将加快系统的响应；但 K_P 过大会使系统有较大的超调，并产生振荡，使稳定性变坏。

增大积分时间 T_I 有利于减小超调，减小振荡，使系统更加稳定，但将延长消除静差的时间。

增大微分时间 T_D 有利于加快系统响应，使超调量减小，稳定性增加；但对扰动有较敏感的反应，减弱了系统对扰动的抑制能力。

可根据经验和不同控制对象的特征，依次整定参数 K_P 、 T_I 、 T_D 。

① 根据控制对象的特点，从表 5.3 找出 PID 各参数的大致范围，按此设置后闭环运行。

② 将比例系数 K_P 由小变大，并观察相应的系统响应，直至得到反应快、超调量小的响应曲线。如果系统静差已在允许范围内，响应曲线也满意，则只需比例调节。

③ 如果比例调节不能满足对系统静差的设计要求，则需加入积分环节。首先置积分时间 T_I 为较大值，并将经整定得到的 K_P 略为缩小（如缩小为原值的 0.8 倍），然后减小积分时间，在保持系统良好动态性能的情况下，消除静差。在此过程中，可根据响应曲线的好坏反复改变 K_P 和 T_I ，以期得到满意的控制过程与整定参数。

④ 若使用 PI 调节消除了静差，但动态过程经反复调整仍不能满意，则可加入微分环节，

构成 PID 调节。整定时先置微分时间 T_D 为 0，然后增大 T_D ，同时相应地改变 K_P 和 T_I ，逐步凑试，以获得满意的调节效果和控制参数。

⑤ 应用 0.618 法、瞎子爬山法、单纯形加速法等优选法，可以大大减少凑试时整定的工作量，或者设计数字 PID 参数自寻最优控制。

表 5.3 PID 各参数的大致范围

被 调 量	特 点	K_P	T_I /分钟	T_D /分钟
流量	时间常数小，并有噪声，故 K_P 较小， T_I 较小，不用微分	1~2.5	0.1~1	
温度	对象有较大滞后，常采用 PID 控制	1.6~0.5	3~10	0.5~3
压力	对象滞后不大，一般不使用微分	1.4~3.5	0.4~3	
液位	如果允许有静差时，可不使用积分和微分	1.25~5		

5.2 纯滞后补偿控制

对许多由于能量或物料的传输延迟而具有纯滞后性质的控制对象而言，抑制超调成为控制系统的主要指标。纯滞后补偿设计的主要思路是，控制不仅要根据目前的偏差，而且要考虑因滞后而影响到目前的过去情况。

5.2.1 大林（Dahlin）算法

1. 原理

在闭环控制系统中（如图 3.8 所示），将具有纯滞后性质的控制对象近似为带纯滞后的一阶或二阶惯性环节 $G_c(s)$ ：

$$G_c(s) = \frac{K}{1 + T_1 s} e^{-\tau s} \tag{5.15}$$

或
$$G_c(s) = \frac{K}{(1 + T_1 s)(1 + T_2 s)} e^{-\tau s} \tag{5.16}$$

大林算法的目标是，设计数字控制器 $D(z)$ ，使闭环传递函数 $\Phi(s)$ 为

$$\Phi(s) = \frac{e^{-\tau s}}{1 + T_\tau s} \tag{5.17}$$

式中， τ 为控制对象纯滞后时间， $\tau = NT$ ，（ N 为正整数）； T_τ 为闭环系统时间常数。

也就是，用零阶保持器法离散化后，可使闭环传递函数 $\Phi(z)$ 为

$$\Phi(z) = \frac{(1 - e^{-T/T_\tau})z^{-N-1}}{1 - e^{-T/T_\tau} z^{-1}} \tag{5.18}$$

对于带纯滞后一阶惯性环节的控制对象，这样设计的数字控制器 $D(z)$ ，其输出可用差分方程表示为

$$u(k) = Ae(k) - Be(k - 1) + Cu(k - 1) + (1 - C)u(k - N - 1) \tag{5.19}$$

可以看出，按大林算法设计的控制器，不但考虑了目前的偏差，将最后两次偏差进行权重分配，而且考虑了 N 次以前的输出情况，滞后越大，则参考值越靠前，因而能有效抑制超调。

2. 带纯滞后一阶惯性对象的大林算法

零阶保持器和纯滞后一阶惯性被控对象连续部分的脉冲传递函数为

$$G(z) = Z \left[\frac{1 - e^{-Ts}}{s} \frac{K}{1 + T_1 s} e^{-NTS} \right] \quad (5.20)$$

$$= Kz^{-N-1} \frac{1 - e^{-T/T_1}}{1 - e^{-T/T_1} z^{-1}}$$

控制器脉冲传递函数 $D(z)$ 为

$$D(z) = \frac{\Phi(z)}{1 - \Phi(z)} \cdot \frac{1}{G(z)} = \frac{(1 - e^{-T/T_1} z^{-1})(1 - e^{-T/T_1})}{K(1 - e^{-T/T_1})[1 - e^{-T/T_1} z^{-1} - (1 - e^{-T/T_1})z^{-N-1}]}$$

$$= \frac{A - Bz^{-1}}{1 - Cz^{-1} - (1 - C)z^{-N-1}} \quad (5.21)$$

离线计算常数为

$$A = \frac{(1 - e^{-T/T_1})}{K(1 - e^{-T/T_1})}$$

$$B = Ae^{-T/T_1}$$

$$C = e^{-T/T_1}$$

化为带纯滞后一阶惯性对象的大林算法的差分表达式(5.19)，即可据此编制程序。

3. 带纯滞后二阶惯性对象的大林算法

设计目标仍然是式(5.16)。零阶保持器和纯滞后二阶惯性被控对象连续部分的脉冲传递函数为

$$G(z) = Z \left[\frac{1 - e^{-Ts}}{s} \frac{K}{(1 + T_1 s)(1 + T_2 s)} e^{-NTS} \right] = Kz^{-N-1} \frac{B_0 + B_1 z^{-1}}{(1 - e^{-T/T_1} z^{-1})(1 - e^{-T/T_2} z^{-1})} \quad (5.22)$$

式中

$$B_0 = 1 + \frac{1}{T_2 - T_1} (T_1 e^{-T/T_1} - T_2 e^{-T/T_2})$$

$$B_1 = e^{-T(1/T_1 + 1/T_2)} + \frac{1}{T_2 - T_1} (T_1 e^{-T/T_2} - T_2 e^{-T/T_1})$$

控制器脉冲传递函数 $D(z)$ 为

$$D(z) = \frac{\Phi(z)}{1 - \Phi(z)} \cdot \frac{1}{G(z)}$$

$$= \frac{(1 - e^{-T/T_1} z^{-1})(1 - e^{-T/T_2} z^{-1})(1 - e^{-T/T_1})}{K(B_0 + B_1 z^{-1})[1 - e^{-T/T_1} z^{-1} - (1 - e^{-T/T_1})z^{-N-1}]}$$

$$(5.23)$$

4. 对象动态特性测取和系统参数整定

(1) 对象动态特性的测取

根据被调量在阶跃输入下的整个变化过程曲线（如图 5.16 所示），可确定对象的纯滞后时间 τ 和时间常数等动态特性。

(2) 参数整定

① 选取采样周期 T ，及 $N = \tau/T$ ；

② 通过实验，将闭环系统时间常数 T_r 调整到最佳：若过小则系统响应快，但易引起振荡；若过大则稳定性好，但响应慢；

③ 确定参数后根据公式计算 A , B , C 等参数。

5. 振铃现象及其消除

(1) 振铃现象分析

振铃 (Ringing) 现象, 是指数字控制器的输出以二分之一采样频率大幅度波动。由于被控对象中惯性环节的低通特性, 这种波动对系统的输出几乎没有影响, 但会增加执行机构的磨损, 在有交互作用的多参数控制系统中还有可能影响稳定性。

单位阶跃输入函数 $R(z) = 1/(1-z^{-1})$, 含有极点 $z=1$ 。如果 $\Phi u(Z) = \Phi(Z)/G(Z) = U(Z)/R(Z)$ 的极点在 z 平面的负实轴上, 且与 $Z=-1$ 点相近, 那么在数字控制器的输出序列 $u(kT)$ 中将含有这两种幅值相近的瞬态项, 而且瞬态项的符号在不同时刻是不相同的: 当两瞬态项符号相同时, 数字控制器的输出控制作用加强; 当符号相反时, 控制作用减弱, 从而造成数字控制器的输出序列大幅度波动。极点离开 $Z=-1$ 点越远, 振铃幅度越小 (如图 5.17 所示)。

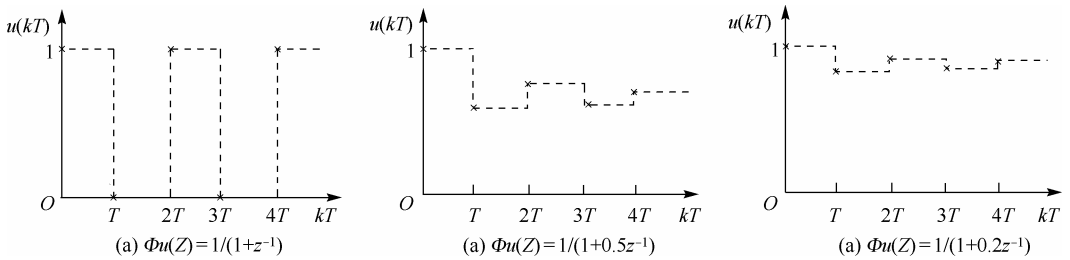


图 5.17 数字控制器输出的波动

通常用振铃幅度 RA (Ringing Amplitude) 表示振铃的强烈程度, 其值等于数字控制器在单位阶跃输入作用下, 第 0 拍输出幅度与第 1 拍输出幅度之差, 即

$$RA = m(0) - m(1) \quad (5.24)$$

在单位阶跃输入作用下, 数字控制器输出的 Z 变换写成

$$U(z) = \frac{1}{1-z^{-1}} \cdot \frac{1+b_1z^{-1}+b_2z^{-2}+\cdots}{1+a_1z^{-1}+a_2z^{-2}+\cdots} = \frac{1+b_1z^{-1}+b_2z^{-2}+\cdots}{1+(a_1-1)z^{-1}+(a_2-a_1)z^{-2}+\cdots} \quad (5.25)$$

$$= 1 + (b_1 - a_1 + 1)z + \cdots$$

$$RA = 1 - (b_1 - a_1 + 1) = a_1 - b_1 \quad (5.26)$$

在带纯滞后的一阶惯性环节组成的系统中, 数字控制器的脉冲传递函数不存在负实轴上的极点, 因而不出现振铃现象。

当被控制对象为带纯滞后的二阶惯性环节时, 在 $\Phi u(z) = \Phi(z)/G(z)$ 的极点中有一个是

$$z = -\frac{B_1}{B_0} \quad (5.27)$$

而

$$\lim_{T \rightarrow 0} \left[-\frac{B_1}{B_0} \right] = -1$$

这说明可能出现负实轴上与 $z=-1$ 相近的极点, 这一极点将引起振铃现象。

(2) 消除振铃现象的方法

① 找出 $D(z)$ 中引起振铃现象的因子 ($z=-1$ 附近的极点), 然后令其中的 $z=1$, 这样处理不影响输出量的稳态值, 但却改变了数字控制器的动态特性, 将影响闭环系统的瞬态性能。

② 选择合适的采样周期 T 及系统闭环时间常数 T_r ，避免产生强烈的振铃现象。例如，在带纯滞后的二阶惯性环节组成的系统中，振铃幅度与被控对象的参数 T_1, T_2 ，闭环系统期望的时间常数 T_r 及采样周期 T 有关。适当选择 T 和 T_r ，可以把振铃幅度抑制在最低限度以内。

5.2.2 史密斯（Smith）预估控制

1. 原理

20 世纪 50 年代就已提出了史密斯预估控制的原理：对纯滞后时间为 τ 的控制对象的控制器，并联一个补偿环节[即史密斯预估控制器 $G_L(s)$]，如图 5.18 所示，使反馈量不受对象滞后的影响，即提前 τ 对系统作用。但模拟仪表无法实现这种补偿，微机控制技术的发展，才使纯滞后补偿可以方便地实现。

若零阶保持器和控制对象连续部分的脉冲传递函数简化为 $G_p(s)e^{-\tau s}$ ， $G_p(s)$ 为不含纯滞后部分， τ 为纯滞后时间。

设计史密斯预估控制器
$$G_L(s) = G_p(s)(1 - e^{-\tau s}) \tag{5.28}$$

则
$$D'(s) = \frac{W(s)}{1 + W(s)G_p(s)(1 - e^{-\tau s})}$$

$$\Phi(s) = \frac{Y(s)}{X(s)} = \frac{D'(s)G_p(s)e^{-\tau s}}{1 + D'(s)G_p(s)e^{-\tau s}} = \frac{W(s)G_p(s)}{1 + W(s)G_p(s)} e^{-\tau s} \tag{5.29}$$

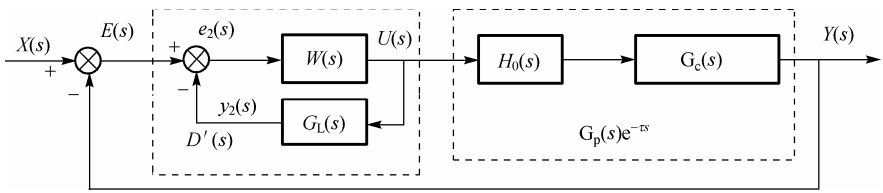


图 5.18 史密斯预估控制原理图

将 $\Phi(s)$ 用图 5.19 表示，可以清楚地看出史密斯预估控制的效果：反馈量不再受对象滞后 $e^{-\tau s}$ 的影响。

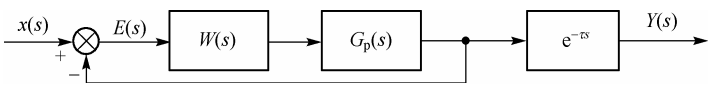


图 5.19 史密斯预估控制效果图

2. 史密斯预估器的计算机实现

设控制对象为一阶惯性纯滞后环节

$$G_c(s) = \frac{K}{1 + T_1 s} e^{-\tau s}$$

考虑到零阶保持器

$$H_0(s) = \frac{1 - e^{-Ts}}{s}$$

$$G_p(s) = \frac{1 - e^{-Ts}}{s} \cdot \frac{K}{1 + T_1 s}$$

史密斯预估控制器的 Z 变换式为

$$\begin{aligned} Z[G_L(s)] &= Z[G_p(s)(1 - e^{-\tau s})] = Z\left[\frac{K(1 - e^{-Ts})}{s(1 + T_1 s)}(1 - e^{-\tau s})\right] \\ &= Z\left[K\left(\frac{1}{s} - \frac{T_1}{1 + T_1 s}\right)(1 - e^{-Ts})(1 - e^{-\tau s})\right] = (1 - z^{-N}) \frac{bz^{-1}}{1 - az^{-1}} \end{aligned} \quad (5.30)$$

式中, $N = \tau/T$; $a = e^{-T/T_1}$; $b = K(1 - e^{-T/T_1})$ 。

根据相应的差分方程, 即可编制程序:

$$y_2(k) = ay_2(k-1) + b[u(k-1) - u(k-N-1)] \quad (5.31)$$

式中, $u(K-1)$, $u(K-N-1)$ 由控制算法 $W(s)$ 决定。

3. 史密斯预估器的缺点

① 预估器对系统受到的负荷干扰无补偿作用。

② 预估控制系统的效果严重依赖于对象的动态模型精度, 特别是纯滞后时间, 模型的失配或运行条件的改变都将影响到控制效果。

5.3 最少拍控制

时间最优控制系统 (例如, 数字随动系统) 除了要求物理可实现性和稳定性外, 还要求快速性: 系统的输出值尽快跟踪给定值的变化; 准确性: 稳定后无静差。

为此, 最少拍控制设计要求闭环系统对于典型输入在最少个采样周期内达到无静差的稳态, 且闭环脉冲传递函数为

$$\Phi(z) = \Phi_1 z^{-1} + \Phi_2 z^{-2} + \dots + \Phi_N z^{-N} \quad (5.32)$$

式中, N 为可能情况下的最小正整数。

最少拍控制的设计与控制对象、输入形式和对输出纹波的要求有关。

5.3.1 不同控制对象的最少拍控制设计

1. 控制对象为一阶惯性环节

设微机闭环控制系统中 (如图 3.8 所示), 零阶保持器和一阶惯性被控对象连续部分的脉冲传递函数简化为

$$G(z) = Z\left[\frac{1 - e^{-Ts}}{s} \frac{1}{1 + s}\right] = (1 - z^{-1}) \left(\frac{1}{1 - z^{-1}} - \frac{1}{1 - e^{-T} z^{-1}}\right) = \frac{(1 - e^{-T}) z^{-1}}{1 - e^{-T} z^{-1}} \quad (5.33)$$

如果输入为单位阶跃函数

$$X(z) = \frac{1}{1 - z^{-1}}$$

则要求一个采样周期 (一拍) 后输出值能跟上输入值的变化。

$$Y(z) = \frac{1}{1-z^{-1}} \cdot z^{-1}$$

$$\Phi(z) = z^{-1}$$

即
应有

$$D(z) = \frac{\Phi(z)}{1-\Phi(z)} \cdot \frac{1}{G(z)} = \frac{1-e^{-T}z^{-1}}{(1-e^{-T})(1-z^{-1})} \quad (5.34)$$

以采样周期为单位时间, 即 $T=1$, 则

$$D(z) = \frac{1-0.368z^{-1}}{0.632(1-z^{-1})} \quad (5.35)$$

此时, $D(z)$ 的输出控制量 $U(z)$ 为

$$\begin{aligned} U(z) &= [X(z) - Y(z)]D(z) = \left[\frac{1}{1-z^{-1}} - \frac{1}{1-z^{-1}} z^{-1} \right] \frac{1-0.368z^{-1}}{0.632(1-z^{-1})} \\ &= 1.582 + z^{-1} + z^{-2} + \dots \\ U(k) &= 1.582, 1, 1, \dots \end{aligned} \quad (5.36)$$

可以看出, 经过一个采样周期, 控制输出稳定为 1, 这样保证了系统的输出 $y(t)$ 无纹波, 实现了控制要求。

2. 控制对象为二阶惯性环节

零阶保持器和二阶惯性被控对象连续部分的脉冲传递函数为

$$\begin{aligned} G(Z) &= Z \left[\frac{1-e^{-Ts}}{s} \frac{K}{s(1+\tau s)} \right] \\ &= K(1-z^{-1}) \left[\frac{Tz^{-1}}{(1-z^{-1})^2} - \frac{\tau}{1-z^{-1}} + \frac{\tau}{1-e^{-T/\tau}z^{-1}} \right] \\ &= K\gamma T \frac{(1+\beta z^{-1})z^{-1}}{(1-z^{-1})(1-\alpha z^{-1})} \end{aligned} \quad (5.37)$$

式中

$$\alpha = e^{-T/\tau}, \quad \gamma = 1 - (1-\alpha)\frac{\tau}{T}, \quad \beta = \frac{1}{\gamma} \left[(1-\alpha)\frac{\tau}{T} - \alpha \right]$$

如果同样在输入为单位阶跃函数时, 要求一拍后输出值能跟上输入值的变化, 设 $T=\tau=1, K=1$,

则

$$\begin{aligned} G(z) &= \frac{0.368z + 0.264}{(z-1)(z-0.368)} \\ D(z) &= \frac{2.717 - z^{-1}}{1 - 0.718Z^{-1}} \end{aligned} \quad (5.38)$$

此时, $D(z)$ 的输出控制量 $U(z)$ 为

$$\begin{aligned} U(z) &= \frac{2.717 - z^{-1}}{1 - 0.718z^{-1}} = 2.717 - 2.95z^{-1} + 2.118z^{-2} - 1.52z^{-3} + 1.09z^{-4} + \dots \\ U(k) &= 2.717, -2.95, 2.118, -1.52, 1.09, \dots \end{aligned} \quad (5.39)$$

控制输出并不稳定，而系统的输出 $y(t)$ 波形如图 5.20 所示。虽然在各采样点偏差为 0，但在采样点之间有纹波。

5.3.2 不同输入函数的最少拍控制设计

1. 典型输入函数

典型输入函数是指

$$x(t) = \frac{1}{(q-1)!} t^{q-1} \quad (5.40)$$

对应的 z 变换为

$$X(z) = \frac{B(z)}{(1-z^{-1})^q} \quad (5.41)$$

式中， $B(z)$ 是不包含 $(1-z^{-1})$ 因子的关于 z^{-1} 的多项式。

当 q 分别等于 1, 2 和 3 时，对应的典型输入为单位阶跃、单位速度和单位加速度输入函数。

2. 设计方法

对于微机闭环控制系统，有

$$E(z) = X(z) - Y(z) = X(z)(1 - \Phi(z))$$

根据 Z 变换终值定理，系统的稳态误差为

$$\lim_{t \rightarrow \infty} e(t) = \lim_{z \rightarrow 1} (1 - z^{-1}) E(z) = \lim_{z \rightarrow 1} (1 - z^{-1}) X(z) (1 - \Phi(z))$$

对于典型输入函数

$$\lim_{t \rightarrow \infty} e(t) = \lim_{z \rightarrow 1} (1 - z^{-1}) \frac{B(z)}{(1 - z^{-1})^q} (1 - \Phi(z))$$

要使稳态误差为 0，就必须使 $(1 - \Phi(z))$ 中包含因子 $(1 - z^{-1})^r$ ，且 $r \geq q$ ，即

$$1 - \Phi(z) = (1 - z^{-1})^r \cdot F(z) \quad r \geq q \quad (5.42)$$

式(5.42)中， $F(z)$ 是由其他条件确定的 z^{-1} 多项式，如果 $G(z)$ 是稳定的且不含纯滞后，为使数字控制器最简单，可取 $F(z) = 1$ 。

$E(z)$ 的 z^{-1} 多项式次数决定系统达到无偏差的时间，为使 $E(z)$ 展开的 z^{-1} 多项式项数最少，即系统达到无偏差的时间最短，应取 $r = q$ 。

3. 单位阶跃、单位速度和单位加速度输入函数的最少拍控制设计

(1) 单位阶跃输入 ($q=1$)

输入为 $x(t) = 1(t)$ ，即

$$X(z) = \frac{1}{1 - z^{-1}}$$

此时

$$\Phi(z) = 1 - (1 - z^{-1}) = z^{-1}$$

则

$$E(z) = X(z)(1 - \Phi(z)) = \frac{1}{1 - z^{-1}} (1 - z^{-1}) = 1$$

$$Y(z) = X(z)\Phi(z) = \frac{1}{1 - z^{-1}} z^{-1} = z^{-1} + z^{-2} + z^{-3} + \dots \quad (5.43)$$

说明只需一个采样周期（一拍），输出就能跟上输入，偏差为 0，即达到稳态，如图 5.21 所示，得出

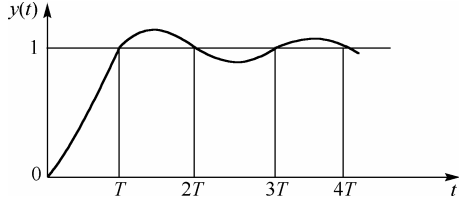


图 5.20 二阶惯性被控对象在阶跃输入时一拍跟上的系统输出波形

$$D(z) = \frac{1}{G(z)} \cdot \frac{\Phi(z)}{1 - \Phi(z)} = \frac{1}{G(z)} \cdot \frac{z^{-1}}{1 - z^{-1}} \quad (5.44)$$

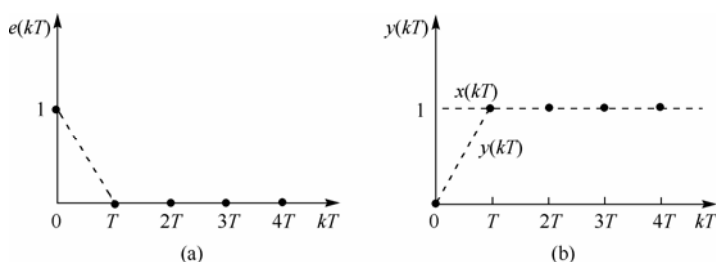


图 5.21 单位阶跃输入时的误差及输出序列

(2) 单位速度输入 ($q=2$)

输入为 $x(t)=t$, 即

$$X(z) = \frac{Tz^{-1}}{(1 - z^{-1})^2}$$

此时

$$\Phi(z) = 1 - (1 - z^{-1})^2 = 2z^{-1} - z^{-2}$$

则

$$E(z) = X(z)(1 - \Phi(z)) = \frac{Tz^{-1}}{(1 - z^{-1})^2} (1 - 2z^{-1} + z^{-2}) = Tz^{-1}$$

$$Y(z) = X(z)\Phi(z) = 2Tz^{-1} + 3Tz^{-2} + 4Tz^{-3} + \dots \quad (5.45)$$

说明只需两个采样周期 (二拍), 就能达到稳态, 如图5.22所示, 得出

$$D(z) = \frac{1}{G(z)} \cdot \frac{\Phi(z)}{1 - \Phi(z)} = \frac{1}{G(z)} \cdot \frac{2z^{-1} - z^{-2}}{(1 - z^{-1})^2} \quad (5.46)$$

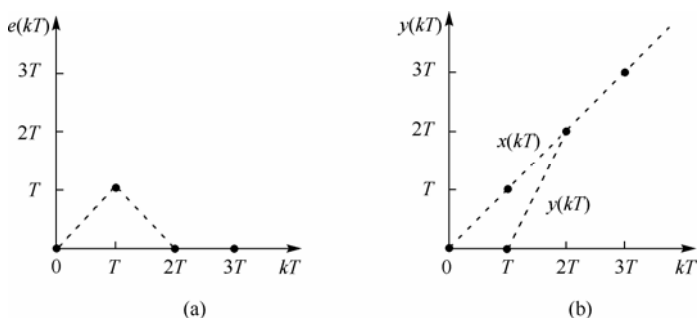


图 5.22 单位速度输入时的误差及输出序列

(3) 单位加速度输入 ($q=3$)

输入为

$$x(t) = \frac{1}{2}t^2$$

即

$$X(z) = \frac{T^2 z^{-1} (1 + z^{-1})}{2(1 - z^{-1})^3}$$

此时

$$\Phi(z) = 1 - (1 - z^{-1})^3 = 3z^{-1} - 3z^{-2} + z^{-3}$$

则

$$E(z) = \frac{1}{2}T^2 z^{-1} + \frac{1}{2}T^2 z^{-2} \quad (5.47)$$

说明只需三个采样周期（三拍），就能达到稳态，如图5.23所示，得出

$$D(z) = \frac{1}{G(z)} \cdot \frac{\Phi(z)}{1-\Phi(z)} = \frac{1}{G(z)} \cdot \frac{3z^{-1} - 3z^{-2} + z^{-3}}{(1-z^{-1})^3} \quad (5.48)$$

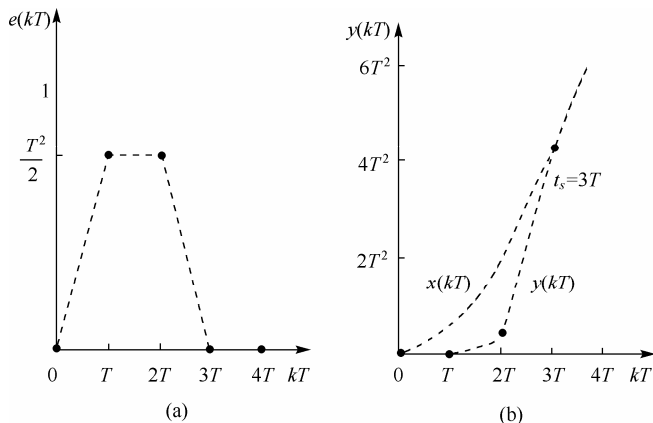


图 5.23 单位加速度输入时的误差及输出序列

5.3.3 最少拍控制设计的局限性

1. 对不同输入的适应性差

最少拍控制器的设计可以使系统对某一典型输入的响应为最少拍，但对于其他典型输入不一定为最少拍，甚至会引起大的超调和静差。也就是说，一种典型的最少拍闭环脉冲传递函数 $\Phi(z)$ 只适应一种特定的输入，而不能适应于各种输入。

一般来说，针对一种典型的输入函数 $X(z)$ 设计，得到的系统闭环脉冲传递函数，当用于次数较低的输入函数 $X(z)$ 时，系统将出现较大的超调，响应时间也会增加，虽然在采样时刻的误差为零；而用于次数较高的输入函数时，输出将不能完全跟踪输入，以致产生稳态误差。

例如，按单位速度输入设计的 $\Phi(z) = 2z^{-1} - z^{-2}$ ，当输入是单位阶跃和单位加速度时，对应的输出分别如图5.24(a)、(b)所示。

$$\begin{aligned} (1) \text{ 输入单位阶跃} \quad X(z) &= \frac{1}{1-z^{-1}} \\ Y(z) &= X(z)\Phi(z) = \frac{1}{1-z^{-1}}(2z^{-1} - z^{-2}) = 2z^{-1} + z^{-2} + z^{-3} + \dots \end{aligned} \quad (5.49)$$

当 $k=1$ 时，超调量达到 100%。

$$\begin{aligned} (2) \text{ 输入单位加速度} \quad X(z) &= \frac{T^2 z^{-1}(1+z^{-1})}{2(1-z^{-1})^3} \\ Y(z) &= X(z)\Phi(z) = \frac{T^2 z^{-1}(1+z^{-1})}{2(1-z^{-1})^3}(2z^{-1} - z^{-2}) \\ &= T^2 z^{-2} + 3.5T^2 z^{-3} + 7T^2 z^{-4} + 11.5T^2 z^{-5} + \dots \end{aligned} \quad (5.50)$$

虽然系统经过二拍达到稳定,但是输入与输出之间始终存在稳态误差,即当 $k \geq 2$ 时, $e(kT) = T^2$ 。

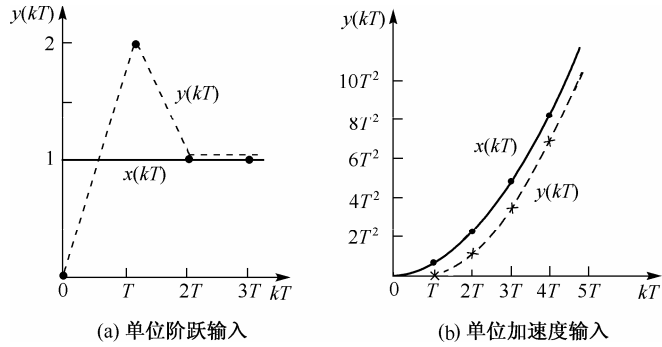


图 5.24 按单位速度输入设计的最少拍控制器对其他输入的响应特性

- 2. 系统输出在采样点之间可能存在纹波

单纯最少拍控制设计的系统,在输出值跟踪输入值后,在采样点之间可能存在纹波(参见图5.15)。原因是数字控制器的输出序列 $u(k)$ 经过若干拍后,不为常值或 0,而是波动的。非采样时刻的纹波现象不仅造成非采样时刻有偏差,而且浪费执行机构的功率,增加机械磨损,因此必须消除。
- 3. 实现性受限制

如果连续被控对象有纯滞后,则所设计的闭环脉冲传递函数 $\Phi(z)$ 中必须含有纯滞后,且滞后时间至少要等于被控对象的滞后时间。否则,要求系统的响应超前被控对象的输入,而这是不可能实现的。

- 4. 稳定性受限制

如果 $G(z)$ 在 z 平面单位圆上和圆外有极点,不满足稳定条件,则需要对设计原则进行相应的限制。

在 $\Phi(z) = \frac{D(z)G(z)}{1 + D(z)G(z)}$ 中,虽然 $D(z)$ 和 $G(z)$ 总是成对出现,但当系统的参数产生漂移,或辨识的参数有误差时,简单地利用 $D(z)$ 的零点去对消 $G(z)$ 中的不稳定极点,是不可能准确实现的,从而将引起闭环系统的不稳定,因此在选择 $\Phi(z)$ 时必须加一个称为稳定性条件的约束。

5.3.4 最少拍无纹波控制的设计

- 1. 必要条件

无纹波系统要求系统的输出信号在采样点之间不出现纹波,必须满足:

 - ① 对阶跃输入,当 $t \geq NT$ 时,有 $y(t) = \text{常数}$;
 - ② 对速度输入,当 $t \geq NT$ 时,有 $y'(t) = \text{常数}$;
 - ③ 对加速度输入,当 $t \geq NT$ 时,有 $y''(t) = \text{常数}$ 。

这样,被控对象 $G(s)$ 中必须含有足够的积分环节,才能给出与系统输入 $x(t)$ 相同且平滑的输出 $y(t)$ 。例如,针对速度输入函数进行设计,稳态过程中 $G(s)$ 的输出必须是速度函数,为了产生这样的速度输出函数, $G(s)$ 中必须至少有一个积分环节,使得控制信号 $u(k)$ 为常值(包括零)时, $G(s)$ 的稳态输出是所要求的速度函数。同样,针对加速度输入函数设计的无纹波控制器, $G(s)$ 中必须至少有两个积分环节。

2. 约束条件

为了消除波纹, 应使 $U(z)/X(z)$ 是 z^{-1} 的有限多项式, 这样在确定的典型输入 $x(t)$ 下, 经过有限节拍后, 控制量 $u(k)$ 就可以达到相对的稳定值。同样 $\Phi(z) = Y(z)/X(z)$ 也应是 z^{-1} 的有限多项式, 在有限节拍后, 输出量 $y(k)$ 可以达到相对的稳定值。

闭环微机控制系统中 (如图3.8 所示),

$$\frac{U(z)}{X(z)} = \frac{\Phi(z)}{G(z)} = \frac{\Phi(z)M_G(z)}{N_G(z)} \quad (5.51)$$

式中, $N_G(z)$ 和 $M_G(z)$ 分别为 $G(z)$ 的零点多项式和极点多项式。

无波纹系统的附加约束条件是, 闭环传递函数 $\Phi(z)$ 必须包含 $G(z)$ 的全部零点, 可形式地表示为

$$\Phi(z) = N_G(z)F(z)$$

式中, $F(z)$ 为待选的有限多项式。

3. 增加调整时间

无波纹系统的调整时间要增加若干拍, 增加的拍数等于 $G(z)$ 在单位圆内的零点数。

例如, 控制对象同样为二阶惯性环节, 在输入仍为单位阶跃函数时, 要求输出值二拍后跟上输入值的变化, 即要求

$$Y(z) = \frac{z}{z-1} z^{-2} + C_1 \frac{z}{z-1} z^{-1} - C_1 \frac{z}{z-1} z^{-2}$$

同样, 设 $T = \tau = 1$, $K = 1$

$$G(z) = \frac{0.368z + 0.264}{(z-1)(z-0.368)}$$

有

$$\Phi(z) = \frac{Y(z)}{X(z)} = \frac{C_1 z + (1 - C_1)}{z^2} \quad (5.52)$$

令 $C_1 = 0.582$

$$Y(z) = \frac{C_1(z-1)+1}{z(z-1)} = \frac{0.582z + 0.418}{z(z-1)}$$

此时, $D(z)$ 的输出控制量 $U(z)$ 为

$$U(z) = \frac{Y(z)}{G(z)} = \frac{1.582z^2 + 0.554z - 0.418}{z^2 + 0.717z} = 1.582 - 0.58z^{-1} + 0 + 0 + \dots \quad (5.53)$$

$$U(k) = 1.582, -0.58, 0, 0, 0, \dots$$

$U(k)$ 经过两个采样周期即稳定为 0, 此系统是无纹波系统。

5.4 串级控制

当被控系统中同时有几个干扰因素可能引起同一个被控量的变化时, 只控制其中一个因素就不能满足系统的控制要求。在单回路控制的基础上, 增加一个或多个控制副回路, 前级调节器的输出作为后级调节器的输入, 构成几个调节器串联而成的串级控制系统, 可以控制其他影响因素, 从而抑制被控对象的时滞特性, 提高系统动态响应的快速性, 显著提高控制效果。

5.4.1 串级控制系统结构和工作原理

计算机串级控制系统框图如图5.25所示，其中外闭环称为主回路或主环，用于最终保证被调量满足工艺要求；内闭环称为副回路或副环，用于克服被控对象所受到的主要干扰。对应有两个调节器：主调节器用于保证系统的控制精度，减小稳态误差，其输入是被调量对给定值的误差，主调节器的输出 $u_1(s)$ 就是副调节器的给定值；副调节器用于保证系统反应灵敏，其输出作为控制命令送到执行机构。如果将副回路看做一个等效对象，则主回路的设计与一般单回路控制系统相似，控制对象称为主控对象，回路中的扰动称为一次扰动；副回路要求时间常数较小，调节通道短，控制对象称为副控对象，回路中的扰动称为二次扰动。

如果选择多个副回路，则构成多串级系统，但无论有多少级，计算的顺序总是由外而内地进行。例如，图5.25所示的双回路串级控制系统在每个采样周期的计算顺序为

- ① 计算主回路的偏差 $e_1(t)$;
- ② 计算主调节器的输出 $u_1(t)$;
- ③ 计算副回路的偏差 $e_2(t)$;
- ④ 计算副调节器的输出 $u_2(t)$ 。

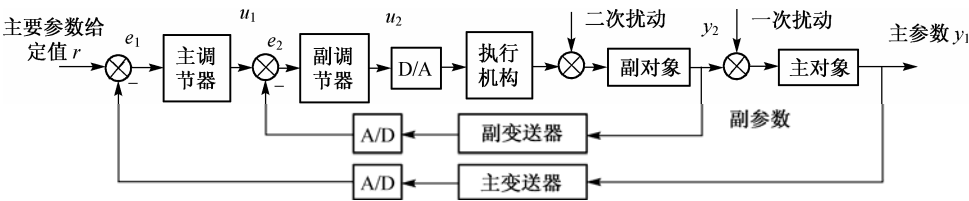


图 5.25 计算机串级控制系统框图

串级控制系统的采样控制方式有两种：

- 同步方式：主副回路的采样控制周期相同；
- 异步方式：主回路的采样控制周期是副回路的整数倍。

5.4.2 串级控制系统的特点

实现原料气加热炉出口温度控制的串级控制系统结构如图5.26所示。

原料气由管道进入加热炉加热，出口原料气的温度 $y_1(t)$ 与给定值 $r(t)$ 比较的误差送入温度调节器，调节器的输出控制阀门的开度，改变燃料油的流量，使加热炉的燃烧状况变化，保证出口原料气的温度 $y_1(t)$ 与 $r(t)$ 一致。

然而，燃料油的压力 $y_2(t)$ 是波动的，燃料油的流量会随着压力的波动而变化，因此压力的波动造成了对出口原料气温度的扰动。由于压力波动到出口原料气温度的变化要经过管道的传输，炉膛的燃烧，加热管道的传热等一系列具有惯性和纯滞后的环节，使得出口原料气温度偏差加大，调节时间加长，用单回路负反馈控制难于获得理想的控制效果。

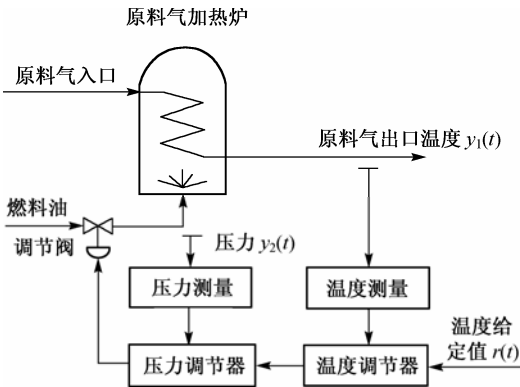


图 5.26 原料气出口温度串级控制系统

设立压力调节副回路可以稳定燃料油的压力 $y_2(t)$ 。由于压力波动经压力测量后直接调节、改变阀位，控制通道的纯滞后很小，惯性也不大，可以及时有效地抑制压力 $y_2(t)$ 的波动。温度调节主回路则用来克服燃料油热值、原料气入口流量、温度、成分的影响，使得控制质量得到显著的提高。

串级控制的方框图如图 5.27 所示。图中，温度调节器和压力调节器分别是主调节器和副调节器。主控对象和副控对象分别是温度和压力。

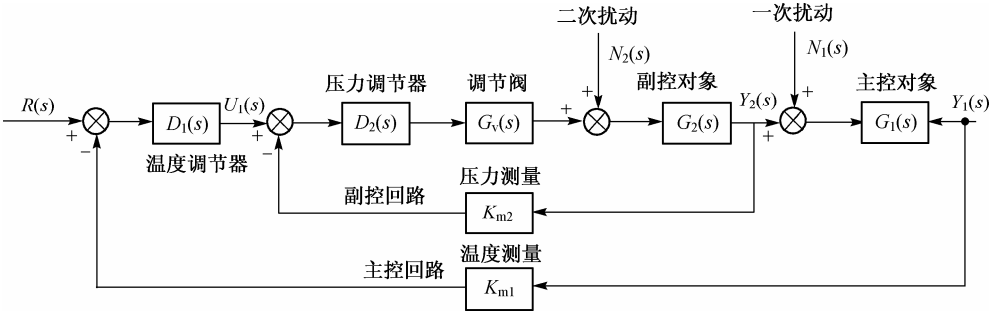


图 5.27 原料气加热炉串级控制系统方框图

为便于分析，设主控对象和副控对象均为一阶惯性环节

$$G_1(s) = \frac{K_1}{1 + T_1 s}, \quad G_2(s) = \frac{K_2}{1 + T_2 s}$$

主调节器和副调节器均为比例控制： $D_1(s) = K_{p1}$ ， $D_2(s) = K_{p2}$ 。调节阀的传递函数 $G_v(s) = K_v$ 。

副控回路传递函数

$$F_2(s) = \frac{Y_2(s)}{U_1(s)} = \frac{K_{p2} K_v \frac{K_2}{1 + T_2 s}}{1 + K_{m2} K_{p2} K_v \frac{K_2}{1 + T_2 s}} = \frac{K'_2}{1 + T'_2 s} \quad (5.54)$$

式中

$$K'_2 = \frac{K_{p2} K_v K_2}{1 + K_{m2} K_{p2} K_v K_2}$$

$$T'_2 = \frac{T_2}{1 + K_{m2} K_{p2} K_v K_2}$$

通常，衡量系统抑制扰动能力的指标为

$$A_{sr} = \frac{Y_1(s)/R(s)}{Y_1(s)/N_2(s)} = D_1(s) D_2(s) G_v(s)$$

现简化为

$$A_{sr} = K_{p1} K_{p2} G_v(s)$$

可以看出：

① 由于 $K_{m2} K_{p2} K_v K_2 \ll 1$ ，得到 $K'_2 < K_2$ ， $T'_2 < T_2$ 。也就是说，减小了副控对象的等效时间常数，对提高系统工作频率有利，使得系统的动作灵敏，反应速度加快，调节更为及时；副控回路放大系数 K'_2 的减小有利于提高主调节器的放大系数 K_{p1} ，提高了系统抑制一次扰动的能力。

② $K_{p1}K_{p2}$ 越大，系统抑制扰动的性能就越好。由于 K_{p2} 的作用，串级控制系统抑制二次扰动的能力比单回路控制高出数十倍乃至上百倍。

③ 负荷变化会引起非线性对象工作点的变化，例如，表现为 K_2 的变化。串级控制系统对负荷和操作条件的变化具有较强的适应能力，原因有两点。

- $K_{m2}K_{p2}K_vK_2 \gg 1$ ，使得副控回路放大系数 K'_2 受 K_2 变化的影响比较小；而 $K'_2 < K_2$ ， K'_2 的减小也减小了负荷变化的影响。
- 串级控制系统的主回路是定值控制系统，副回路是随动控制系统，副调节器的给定值可以由主调节器按照操作条件和负荷变化相应地调整，从而保证在负荷和操作条件发生变化的情况下，调节系统仍然具有较好的品质。

5.4.3 串级控制系统的设计

1. 设计原则

① 把系统中主要的扰动包含在副控回路中，这样，可以在扰动影响到主控参数之前，就大大削弱其影响。

② 副控回路应该尽量包含积分环节，以减小积分环节引起的相角滞后，有利于改善调节系统的品质。

③ 必须用一个可以观测的中间变量作为副控被调参数。

④ 采用异步采样控制方式时，使主回路的采样控制周期比副回路大三倍以上，以避免控制回路之间发生相互干扰和共振。

⑤ 主控调节器 $D_1(z)$ ，为了减小稳态误差，提高控制精度，应该具有积分控制，为了反应灵敏，动作迅速，应该加入微分控制，因此，串级主控调节器 $D_1(z)$ 应该具有 PID 调节规律。

⑥ 副调节器 $D_2(z)$ 较少采用 PID 调节，通常采用比例控制，当副控调节器的比例系数不能太大时（例如，流量、压力控制系统），应当加入积分控制，即采用 PI 调节规律。

⑦ 副调节器也可以按照预期闭环特性 $F_2(z)$ 来设计。

$$F_2(z) = \frac{Y_2(z)}{U_1(z)} = \frac{D_2(z)HG_2(z)}{1 + D_2(z)HG_2(z)} \tag{5.55}$$

式中 $HG_2(z) = Z[H_0(s)G_2(s)]$

如果要求 $F_2(z) = 1$ ，也就是要求输出量 $Y_2(z)$ 随时等于输入，但对于具有惯性或滞后的对象，这意味着当系统受到扰动或者负荷变化时，控制装置必须提供给对象以无限大的能量。这在物理上是无法实现的。

通常选择

$$F_2(z) = z^{-n} \tag{5.56}$$

式中， n 是 $HG_2(z)$ 分子或分母有理多项式的最高幂次。

例如，当对象为一阶惯性时，系统达到稳态的时间可选为一个采样周期；当对象为二阶惯性时，系统达到稳态的时间可选为两个采样周期。

2. 副回路微分先行串级控制系统

副回路微分先行就是在副回路的反馈通道中加入微分控制，可以防止主控制器输出（也

就是副控制器的给定值) 过大而引起副回路的不稳定, 也可以克服副对象惯性较大而引起调节品质的恶化。系统结构如图5.28所示。

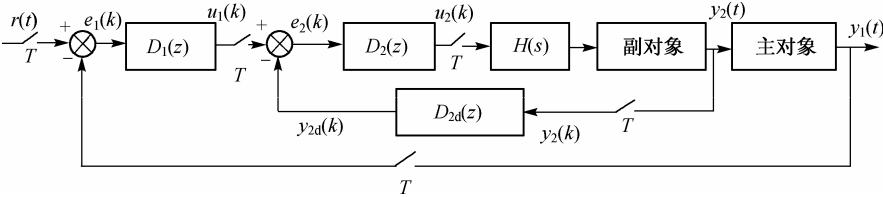


图 5.28 副回路微分先行串级控制系统

图 5.28 中, 微分先行环节的微分方程为

$$\alpha T_2 \frac{dy_{2d}(t)}{dt} + y_{2d}(t) = T_2 \frac{dy_2(t)}{dt} + y_2(t) \tag{5.57}$$

相应差分表达式为

$$y_{2d}(k) = \phi_1 y_{2d}(k-1) + \phi_2 y_2(k) - \phi_3 y_2(k-1) \tag{5.58}$$

式中

$$\phi_1 = \frac{\alpha T_2}{\alpha T_2 + T}, \quad \phi_2 = \frac{T_2 + T}{\alpha T_2 + T}, \quad \phi_3 = \frac{T_2}{\alpha T_2 + T}$$

副回路微分先行串级控制算法的顺序和一般串级控制系统相似, 不同点只是在计算主调节器的输出 u_1 后, 多一步依据微分先行环节的差分表达式计算其输出 $y_{2d}(k)$, 然后再计算副回路的偏差

$$e_2(k) = u_1(k) - y_{2d}(k) \tag{5.59}$$

3. 多回路串级控制系统

双回路的分析方法可以推广应用到多回路串级控制系统, 算法步骤也类似, 运算也是从最外面的回路开始, 逐渐向里面推进; 遇到分叉点, 则先处理优先级较高的回路。与串级控制系统的采样周期相比, 微机的运算速度要快得多, 完全可以忽略各个回路计算时间的先后之别。

5.4.4 串级控制系统的应用

串级控制系统减小了副控对象的等效时间常数, 系统工作频率得以提高, 提高了系统抑制扰动的能力和对负荷和操作条件变化的适应能力, 因而控制性能比单回路控制系统更好。通常, 串级控制系统的作用如下。

- 1. 抑制系统的主要干扰**
通常副回路抑制干扰的能力比单回路控制高出十几倍乃至上百倍, 将主要扰动置于副回路中, 主参数可获得更强的抑制干扰的能力。
- 2. 克服对象的纯滞后**
当对象的纯滞后比较大时, 若采用单回路控制, 则过渡时间较长, 超调量较大, 主参数的控制质量较差。由于串级控制中副调节器的作用, 可以减小等效时间常数, 提高工作频率。
- 3. 减小对象的非线性影响**
对于具有非线性的对象, 设计时将非线性尽可能地包含在副回路中, 由于副回路是随动系统, 能够适应操作条件和负荷的变化, 自动改变给定值, 因此具有一定的自适应能力。

5.5 前 馈 控 制

5.5.1 前馈控制原理与结构

1. 前馈控制原理

反馈控制是偏差控制，前提是被控量在某种干扰的作用下偏离给定值，然后再抑制干扰的影响。如果干扰不断地作用，则系统将出现波动，尤其是被控对象滞后大时，波动就更为严重。

前馈控制则是“弭风于青萍之末”。在某个干扰量影响到被控对象之前就对它直接产生校正作用，来抵消它的影响，是一种针对某一特定干扰实施控制作用的开环控制形式。通常采用前馈与其他控制相结合的方案，当控制算法与参数选择适当时，可以取得很好的效果。

仍以原料气加热炉温度控制系统为例，前馈控制环节如图 5.29 所示。若系统中入口原料气流量有扰动时，由于系统的惯性和纯滞后，控制系统仍然不能保证出口原料气的温度平稳。加入前馈控制，由前馈调节器依据入口原料气流量 $n(t)$ ，按一定规律控制调节阀，使燃料油的流量改变，从而保证了出口原料气温度的稳定。

2. 前馈控制典型结构

前馈控制的典型结构如图 5.30 所示。

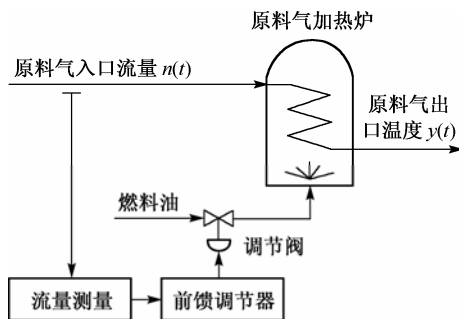


图 5.29 原料气加热炉前馈控制环节

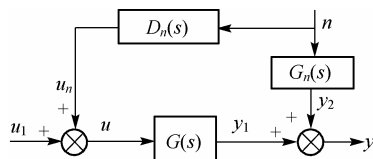


图 5.30 前馈控制的典型结构

图中设 $G_n(s)$ 是被控对象扰动通道的传递函数； $D_n(s)$ 是前馈控制器的传递函数； $G(s)$ 是被控对象控制通道的传递函数； n ， u ， y 分别为扰动量、控制量、被控量。

若要使扰动引起的被控量变化为零，则前馈应起完全补偿扰动作用，即

$$D_n(s)G(s) + G_n(s) = 0$$

前馈控制器的传递函数为

$$D_n(s) = -\frac{G_n(s)}{G(s)} \quad (5.60)$$

5.5.2 前馈-反馈控制

前馈与反馈控制相结合的控制结构，既能发挥前馈控制对主要扰动的补偿作用，又能保留反馈控制对偏差的控制作用，具有既及时又精确的优点。

如图 5.31 所示，在反馈控制的基础上，增加一个扰动的前馈控制，就构成前馈-反馈，完全补偿的条件仍然不变。由式(5.60)，则

$$D_n(s) = \frac{u_n(s)}{N(s)} = -\frac{G_n(s)}{G(s)}$$

得出前馈调节器的微分方程和可据以编写程序的差分方程。

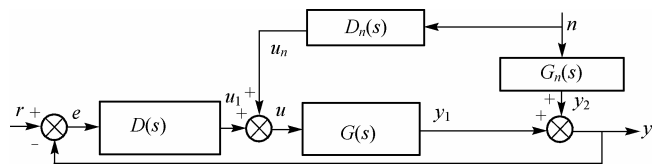


图 5.31 前馈-反馈控制结构

5.5.3 前馈-串级控制

前馈-串级控制（如图5.32所示）能及时克服进入前馈回路和串级副回路的干扰对被控量的影响，前馈控制的输出不是直接作用于执行机构，而是补充到串级控制副回路的给定值中，从而降低了对执行机构动态响应性能的要求，可以更好地改善系统性能。

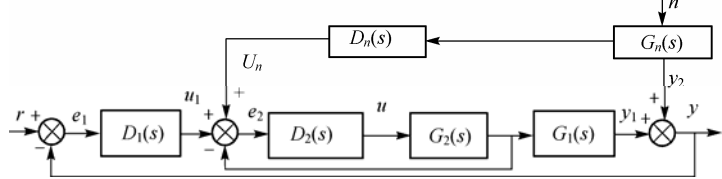


图 5.32 前馈-串级系统结构

原料气加热炉的前馈-串级控制如图5.33所示。在串级控制的基础上，引入前馈控制，以克服入口原料气流量 $n(t)$ 波动的影响。流量测量环节把扰动量的大小输送到前馈调节器，前馈调节器的输出送到比较器。

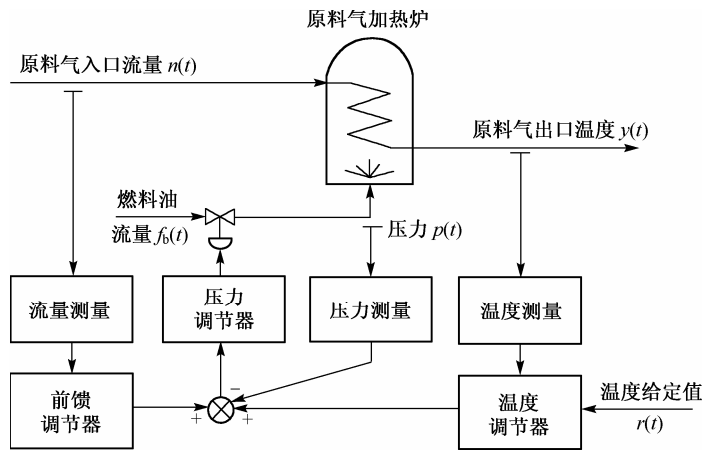


图 5.33 原料气加热炉前馈-串级控制

5.5.4 前馈控制的设计原则与参数整定

1. 设计原则

① 当系统中存在着幅度大、频率高、可测不可控、对被控参数的影响显著的扰动，反馈控制又难以消除其对被控参数的影响时，可引入前馈控制。

② 实际工程中，往往控制通道和扰动通道的惯性时间和纯滞后时间都比较接近，采用静态前馈，即将前馈调节器简化为静态前馈系数 K_f ，通常就能满足控制要求。

③ 当主要扰动无法包围在串级控制的副控回路时，采用前馈-反馈控制可以获得较好的控制效果。

④ 当扰动通道的时间常数远远大于控制通道的时间常数时，反馈控制已能获得良好的控制性能；只有当控制性能要求很高时，才有必要引入前馈控制。

⑤ 当扰动通道的时间常数远远小于控制通道的时间常数时，由于扰动的影响十分快速，前馈调节器的输出迅速达到最大或最小，以至于难以补偿扰动的影响，此时前馈控制的作用不大。

2. 参数整定

前馈调节器的控制规律取决于对象特性数学模型的精确度，然而，实际的对象特性的理论推导和实验测试总难免误差，工程整定方法是很有价值的。静态前馈系数的整定通常有开环整定法、闭环整定法和前馈-反馈整定法。

(1) 开环整定法

开环整定时，系统首先接成负反馈控制，使 $y(t) = r(t)$ ，由工艺分析决定静态前馈系数 K_f 的正负符号。

断开负反馈回路，在系统开环的情况下，施加阶跃扰动， K_f 逐渐由零加大，直到 $y(t) = r(t)$ ，此时的 K_f 即为静态前馈系数整定值。

开环整定法只适用于不存在其他占主要地位扰动的系统，否则得不到正确的结果。当断开负反馈时，必须注意防止被控参数大幅度偏离给定值，避免导致事故。

(2) 闭环整定法

系统工作在闭环状态，稳定后记下扰动的稳态值 $n_0(t)$ 和反馈调节器的输出 $u_0(t)$ 。进行阶跃扰动，待系统稳定以后，记下扰动量 $n_1(t)$ 和反馈调节器的输出 $u_1(t)$ 。

静态前馈系数
$$K_f = \frac{u_1(t) - u_0(t)}{n_1(t) - n_0(t)}$$

(3) 前馈-反馈整定法

在负反馈系统中引入静态前馈 K_f 。进行阶跃扰动，逐步加大 K_f 。反复调试，直到获得比较满意的过渡过程为止。

(4) 动态前馈调节器参数的整定

假设动态前馈调节器
$$D_f(s) = K_f \frac{1 + T_2 s}{1 + T_1 s}$$

令 $T_1 = T_2 = 0$ ，用静态前馈系数整定的方法确定 K_f 。设置 T_1 为某值，逐渐改变 T_2 值，使过渡过程特性调到最好。固定 T_2 值，逐渐改变 T_1 ，再次调整过渡过程特性。多次反复调整 T_1 ， T_2 ，直到控制性能达到要求。

5.6 线性数字控制器的计算机实现

5.6.1 线性数字控制器的一般形式

线性数字控制器一般可表示为

$$D(z) = \frac{U(z)}{E(z)} = \frac{b_0 + b_1 z^{-1} + \cdots + b_m z^{-m}}{1 + a_1 z^{-1} + \cdots + a_n z^{-n}} = \frac{\sum_{i=0}^m b_i z^{-i}}{1 + \sum_{i=1}^n a_i z^{-i}} \quad (5.61)$$

为了保证物理可实现，应使 $D(z)$ 式中不出现未来时刻的信号，即

$$m \geq n$$

为了及时起到控制作用，应使 $D(z)$ 本身不延时，即

$$b_0 \neq 0$$

因此， $D(z)$ 的最简结构为

$$D(z) = \frac{b_0 + b_1 z^{-1} + \cdots + b_m z^{-m}}{1 - z^{-1}} \quad (5.62)$$

当 $m=0$ 、 $n=0$ 时，式(5.61)就是比例控制的表达式

$$D(z) = b_0$$

$$U(z) = b_0 E(z) \quad (5.63)$$

相应的差分表达式为

$$u(k) = b_0 e(k) \quad (5.64)$$

当 $m=2$ 、 $n=1$ 时，式(5.61)就是 PID 控制的表达式为

$$D(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 - z^{-1}}$$

$$U(z) = U(z) z^{-1} + b_0 E(z) + b_1 E(z) z^{-1} + b_2 E(z) z^{-2} \quad (5.65)$$

相应的差分表达式[可与式(5.7)对比]为

$$u(k) = u(k-1) + b_0 e(k) + b_1 e(k-1) + b_2 e(k-2) \quad (5.66)$$

5.6.2 数字控制算法的计算机实现

在微机控制系统中，数字控制算法 $D(z)$ 常用的实现方法有：

1. 直接实现

由式(5.61)所示的一般形式，交叉相乘后转换为差分形式

$$\begin{aligned} u(k) &= b_0 e(k) + b_1 e(k-1) + \cdots + b_m e(k-m) - a_1 u(k-1) - a_2 u(k-2) - \cdots - a_n u(k-n) \\ &= \sum_{i=0}^m b_i e(k-i) - \sum_{i=1}^n a_i u(k-i) \end{aligned} \quad (5.67)$$

即可据此编制程序。

利用递归等编程技巧和循环队列等数据结构，可以减少移位操作次数和占用存储单元。

2. 串联实现

$D(z)$ 具有较高阶次时，可以化作如下一阶或二阶环节 $D_i(z)$ 的串联。

$$D(z) = d_0 \prod_{i=1}^l D_i(z) \quad (1 \leq l \leq n) \quad (5.68)$$

式中

$$D_i(z) = \frac{U_i(z)}{E_i(z)} = \frac{1 + \beta_i z^{-1}}{1 + \alpha_i z^{-1}}$$

或

$$D_i(z) = \frac{U_i(z)}{E_i(z)} = \frac{1 + \beta_{1i} z^{-1} + \beta_{2i} z^{-2}}{1 + \alpha_{1i} z^{-1} + \alpha_{2i} z^{-2}}$$

其实际物理意义就是一系列低阶控制器的串联，如图5.34所示。

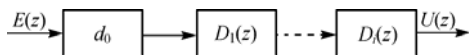


图 5.34 $D(z)$ 的串联实现

3. 并联实现

具有较高阶次的 $D(z)$ 还可以化作如下一阶或二阶环节 $D_i(z)$ 的并联。

$$D(z) = d_0 + \sum_{i=1}^l D_i(z) \quad (1 \leq l \leq n) \quad (5.69)$$

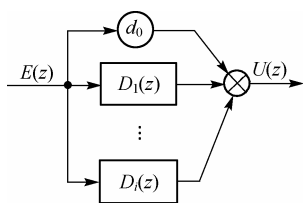


图 5.35 $D(z)$ 的并联实现

其实际物理意义就是一系列低阶控制器的并联，如图 5.35 所示。

串联或并联法通过重复调用一阶或二阶控制器子程序便可实现高阶数字控制器 $D(z)$ ，简化了程序设计，程序占用内存较少，调试和阅读都比较容易。当然，只有在 $D(z)$ 的零点和极点已知的情况下，将高阶的 Z 传递函数分解为一阶、二阶环节才是简便的。

习题与思考题

- 1 试比较数字 PID 位置型控制算法和增量型控制算法的优缺点。
- 2 PID 参数 T , K_p , T_i , T_D 对系统的特性有什么影响？以一种工程整定法为例，说明确定这些参数的步骤。
- 3 数字 PID 控制算法可以实现哪些模拟调节器不具备的功能？
- 4 PID 控制采样周期的选择需要考虑哪些因素？
- 5 根据兼顾干扰抑制和目标跟随性的 PID 控制基本思路，进行离散化设计并编制程序流程图。
- 6 设模拟调节器的传递函数为

$$D(s) = \frac{U(s)}{E(s)} = \frac{1 + 0.2s}{1 + 0.1s}$$

若取采样周期 $T = 0.2$ s，试写出相应数字控制器的增量型控制程序。

- 7 被控对象的传递函数为

$$G_c(s) = \frac{10}{s+1} e^{-s}$$

采样周期 $T = 1$ s。要求：

- (1) 采用大林算法设计数字控制器 $D(z)$ ，并求取控制器的输出 $u(k)$ ；
- (2) 采用史密斯预估算法设计数字控制器 $D'(z)$ ，并求取控制器的输出 $u(k)$ ，设 $W(s) = k'$ 。

8 已知被控对象的传递函数为

$$G_c(s) = \frac{10}{s(0.1s + 1)}$$

采样周期 $T=0.1\text{ s}$ ，采用零阶保持器，针对单位速度输入信号，设计最少拍无纹波系统的 $D(z)$ ，并计算输出响应 $y(k)$ 、控制信号 $u(k)$ 和误差 $e(k)$ 序列，画出它们对时间变化的波形。

9 在图5.28所示的副回路微分先行串级控制系统中，已知采样周期为 T ，且有

$$D_1(z) = \frac{a_0 - a_1 z^{-1} + a_2 z^{-2}}{1 - z^{-1}}, \quad D_2(z) = \frac{b_0 - b_1 z^{-1}}{1 - z^{-1}}$$

常数 $a_0, a_1, a_2, b_0, b_1, \alpha, T_2$ 可分别使 $D_1(z), D_2(z), D_{2d}(z)$ 实现，试写出系统算法。

10 设被控对象的扰动通道传递函数为 $G_n(s) = \frac{K_1}{1 + T_1 s} e^{-\tau_1 s}$ ，控制通道传递函数为 $G(s) = \frac{K_2}{1 + T_2 s} e^{-\tau_2 s}$ ，试推导前馈控制器的传递函数 $D_n(s)$ 及其差分表达式。

第 6 章 现代控制技术

经典控制理论，利用拉普拉斯变换和 Z 变换、脉冲传递函数、方块图等工具，分析和设计单输入单输出的线性系统，其优点是简洁明了，物理概念清楚，易于被人们所接受。但是除简单情况之外，经典方法不适用于非线性系统和多变量系统。事实上，许多现代控制理论和设计技术都是以系统状态变量模型为基础的。

6.1 线性离散系统的状态空间描述

建立在矩阵理论基础上的状态空间法，可以对多输入多输出（MIMO）系统进行描述、分析和设计，更方便地应用于复杂的多变量系统、非线性系统和时变系统。

6.1.1 MIMO线性离散系统的状态空间描述

在连续控制系统中，状态空间分析法是分析、研究系统的有力工具，它解决了频率特性法解决不了的问题。对于离散系统，同样可以用离散状态空间分析法来研究和分析。与 Z 变换法相比，离散状态空间分析法有以下优点：

- ① 采用状态变量法有利于直接利用计算机求解和分析；
- ② 状态变量法不仅适用于单输入单输出系统，也适用于多变量系统，并且在各种情况下系统模型具有统一的形式；
- ③ 状态变量法还可应用于非线性系统、时变系统的分析与设计；
- ④ 状态变量法有利于采用现代分析的方法，如优化方法等实现控制系统设计。

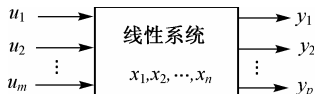


图 6.1 线性连续系统的变量关系

在线性连续系统中，用控制变量 u_m ，状态变量 x_n 和输出变量 y_p 来表征系统的动态特性，如图 6.1 所示。状态变量 x_j 是表征系统本身特性的，并且是变量个数最少的一组变量，变量的个数就是系统的阶数。系统的状态变量 x_j 可以有多种选择方案。

状态变量可以表示成 $n \times 1$ 列向量

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_n(t) \end{bmatrix} \quad (6.1)$$

控制变量可以表示成， $m \times 1$ 向量

$$\mathbf{u}(t) = \begin{bmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_m(t) \end{bmatrix} \quad (6.2)$$

输出变量可以表示成 $p \times 1$ 列向量

$$\mathbf{y}(t) = \begin{bmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_p(t) \end{bmatrix} \quad (6.3)$$

线性连续系统的状态空间表达式为

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) \quad (6.4)$$

$$\mathbf{x}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}u(t) \quad (6.5)$$

式中, $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ 是定常的系数矩阵。

式(6.4)称为状态方程, 式(6.5)称为输出方程。

与线性连续系统类似, 线性离散系统的离散状态空间表达式可以表示为

$$\mathbf{x}(kT + T) = \mathbf{F}\mathbf{x}(kT) + \mathbf{G}u(kT) \quad (6.6)$$

$$\mathbf{y}(kT) = \mathbf{C}\mathbf{x}(kT) + \mathbf{D}u(kT) \quad (6.7)$$

式(6.6)称为状态方程, 式(6.7)称为输出方程。

\mathbf{F} 是 $n \times n$ 维矩阵, 称为状态矩阵或系统矩阵; \mathbf{G} 是 $m \times n$ 维矩阵, 称为输入矩阵或驱动矩阵; \mathbf{C} 是 $p \times n$ 维矩阵, 称为输出矩阵; \mathbf{D} 是 $p \times m$ 维矩阵, 称为直传矩阵或传输矩阵。

线性离散系统的状态变量图如图6.2所示。

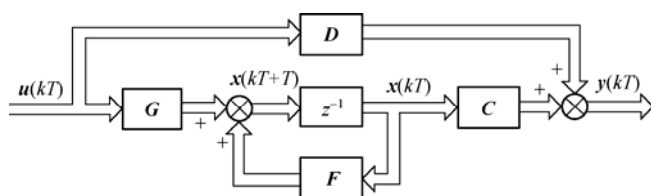


图 6.2 线性离散系统的状态变量图

1. 由差分方程导出离散状态空间表达式

对于单输入—单输出的线性离散系统, 可以用 z 阶差分方程来描述:

$$\begin{aligned} y(kT + nT) + a_1 y(kT + nT - T) + \cdots + a_n y(kT) \\ = b_0 u(kT + mT) + b_1 u(kT + mT - T) + \cdots + b_m u(kT) \end{aligned} \quad (6.8)$$

或表示成

$$y(kT + nT) = \sum_{i=0}^m b_i u(kT + mT - iT) + \sum_{i=1}^n a_i y(kT + nT - iT) \quad (6.9)$$

为了得到状态空间表达式, 应适当选择状态变量, 将高阶的差分方程化为一阶差分方程组, 然后表示成向量的形式, 得到离散状态空间表达式。

(1) $m=0$, 即控制变量不包含高于一阶的差分

$$y(kT + nT) + a_1 y(kT + nT - T) + \cdots + a_n y(kT) = b_0 u(kT) \quad (6.10)$$

选择状态变量为

$$\begin{cases} x_1(kT) = y(kT) \\ x_2(kT) = y(kT + T) \\ x_3(kT) = y(kT + 2T) \\ \vdots \\ x_n(kT) = y(kT + nT - T) \end{cases} \quad (6.11)$$

由式(6.11)可得

$$\begin{cases} x_1(kT + T) = y(kT + T) = x_2(kT) \\ x_2(kT + T) = y(kT + 2T) = x_3(kT) \\ x_3(kT + T) = y(kT + 3T) = x_4(kT) \\ \vdots \\ x_n(kT + T) = y(kT + nT) \\ \quad = -a_n x_1(kT) - a_{n-1} x_2(kT) - \cdots - a_1 x_n(kT) + b_0 u(kT) \end{cases} \quad (6.12)$$

写成矩阵形式

$$\begin{bmatrix} x_1(kT + T) \\ x_2(kT + T) \\ \vdots \\ x_n(kT + T) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ -a_n & -a_{n-1} & -a_{n-2} & \cdots & -a_1 \end{bmatrix} \begin{bmatrix} x_1(kT) \\ x_2(kT) \\ \vdots \\ x_n(kT) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ b_0 \end{bmatrix} u(kT) \quad (6.13)$$

$$[y(kT)] = [1 \quad 0 \quad 0 \quad \cdots \quad 0] \begin{bmatrix} x_1(kT) \\ x_2(kT) \\ \vdots \\ x_n(kT) \end{bmatrix} \quad (6.14)$$

式(6.13), 式(6.14)表示成向量形式为

$$\begin{cases} \mathbf{x}(kT + T) = \mathbf{F}\mathbf{x}(kT) + \mathbf{G}u(kT) \\ \mathbf{y}(kT) = \mathbf{C}\mathbf{x}(kT) \end{cases} \quad (6.15)$$

式中, 状态矩阵为

$$\mathbf{F} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ -a_n & -a_{n-1} & -a_{n-2} & \cdots & -a_1 \end{bmatrix} \quad (6.16)$$

输入矩阵为

$$\mathbf{G} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ b_0 \end{bmatrix} \quad (6.17)$$

输出矩阵为

$$\mathbf{C} = [1 \quad 0 \quad 0 \quad \cdots \quad 0] \quad (6.18)$$

直传矩阵为

$$\mathbf{D} = [0] \quad (6.19)$$

【例 6.1】线性离散系统的差分方程为

$$y(kT + 4T) + 3y(kT + 3T) + 5y(kT + 2T) + 4y(kT + T) + 6y(kT) = 2u(kT)$$

试导出离散状态空间表达式。

解：由差分方程可知：\$n = 4\$, \$m = 0\$, \$p = 1\$（输出向量维数）。

由 \$a_0 = 1\$, \$a_1 = 3\$, \$a_2 = 5\$, \$a_3 = 4\$, \$a_4 = 6\$, \$b_0 = 2\$, 可知

$$\mathbf{F} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -6 & -4 & -5 & -3 \end{bmatrix} \quad \mathbf{G} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 2 \end{bmatrix} \quad \mathbf{C} = [1 \quad 0 \quad 0 \quad 0]$$

离散状态空间表达式为

$$\begin{bmatrix} x_1(kT + T) \\ x_2(kT + T) \\ x_3(kT + T) \\ x_4(kT + T) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -6 & -4 & -5 & -3 \end{bmatrix} \begin{bmatrix} x_1(kT) \\ x_2(kT) \\ x_3(kT) \\ x_4(kT) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 2 \end{bmatrix} u(kT)$$

$$\mathbf{y}(kT) = [1 \quad 0 \quad 0 \quad 0] \begin{bmatrix} x_1(kT) \\ x_2(kT) \\ x_3(kT) \\ x_4(kT) \end{bmatrix}$$

(2) \$m \neq 0\$, 即控制变量包含高于一阶的差分

$$\begin{aligned} & y(kT + nT) + a_1 y(kT + nT - T) + \cdots + a_n y(kT) \\ & = b_0 u(kT + nT) + b_1 u(kT + nT - T) + \cdots + b_n u(kT) \end{aligned} \quad (6.20)$$

选择状态变量

$$\begin{cases} x_1(kT) = y(kT) - h_0 u(kT) \\ x_2(kT) = x_1(kT + T) - h_1 u(kT) \\ x_3(kT) = x_2(kT + T) - h_2 u(kT) \\ \vdots \\ x_n(kT) = x_{n-1}(kT + T) - h_{n-1} u(kT) \end{cases} \quad (6.21)$$

式中

$$\begin{cases} h_0 = b_0 \\ h_1 = b_1 - a_1 h_0 \\ h_2 = b_2 - a_1 h_1 - a_2 h_0 \\ \vdots \\ h_n = b_n - a_1 h_{n-1} - a_2 h_{n-2} - \cdots - a_n h_0 \end{cases} \quad (6.22)$$

系统的离散状态空间表达式为

$$\begin{cases} \mathbf{x}(kT + T) = \mathbf{F}\mathbf{x}(kT) + \mathbf{G}u(kT) \\ \mathbf{y}(kT) = \mathbf{C}\mathbf{x}(kT) + \mathbf{D}u(kT) \end{cases} \quad (6.23)$$

$$\mathbf{F} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ -a_n & -a_{n-1} & -a_{n-2} & \cdots & -a_2 & -a_1 \end{bmatrix} \quad \mathbf{G} = \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_{n-1} \\ h_n \end{bmatrix}$$

$$\mathbf{C} = [1 \quad 0 \quad 0 \quad \cdots \quad 0 \quad 0] \quad \mathbf{D} = [h_0] = [b_0]$$

系统的初始条件 $x_1(0), x_2(0), \cdots, x_n(0)$ 可由下式决定:

$$\begin{cases} x_1(0) = y(0) - h_0 u(0) \\ x_2(0) = y(T) - h_0 u(T) - h_1 u(0) \\ x_3(0) = y(2T) - h_0 u(2T) - h_1 u(T) - h_2 u(0) \\ \vdots \\ x_n(0) = y(nT - T) - h_0 u(nT - T) - h_1 u(nT - 2T) - \cdots - h_{n-2} u(T) - h_{n-1} u(0) \end{cases} \quad (6.24)$$

【例 6.2】设线性离散系统的差分方程为

$$y(kT + 2T) + y(kT + T) + 0.16y(kT) = u(kT + T) + 2u(kT)$$

试写出离散状态空间表达式。

解: 设状态变量

$$x_1(kT) = y(kT)$$

$$x_2(kT) = x_1(kT + T) - u(kT)$$

则

$$\begin{cases} x_1(kT + T) = x_2(kT) + u(kT) \\ x_2(kT + T) = -0.16x_1(kT) - x_2(kT) + u(kT) \\ y(kT) = x_1(kT) \end{cases}$$

系统的离散状态空间表达式为

$$\begin{bmatrix} x_1(kT + T) \\ x_2(kT + T) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -0.16 & -1 \end{bmatrix} \begin{bmatrix} x_1(kT) \\ x_2(kT) \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u(kT)$$

$$\mathbf{y}(kT) = [1 \quad 0] \begin{bmatrix} x_1(kT) \\ x_2(kT) \end{bmatrix}$$

初始条件可由下式决定:

$$\begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} = \begin{bmatrix} y(0) \\ y(T) - u(0) \end{bmatrix}$$

此题也可以直接利用公式求出 $\mathbf{F}, \mathbf{G}, \mathbf{C}$, 从而得到离散状态空间表达式。

线性离散系统得阶数 $n=2$,

$$a_0 = 1, a_1 = 1, a_2 = 0.16$$

$$b_0 = 0, b_1 = 1, b_2 = 2$$

$$h_0 = b_0 = 0$$

$$h_1 = b_1 - a_1 h_0 = 1$$

$$h_2 = b_2 - a_1 h_1 - a_2 h_0 = 1$$

$$\begin{aligned} \mathbf{F} &= \begin{bmatrix} 0 & 1 \\ -0.16 & -1 \end{bmatrix} & \mathbf{G} &= \begin{bmatrix} h_1 \\ h_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \\ \mathbf{C} &= [1 \quad 0] & \mathbf{D} &= [h_0] = [b_0] = 0 \end{aligned}$$

$$\begin{cases} \mathbf{x}(kT+T) = \begin{bmatrix} 0 & 1 \\ -0.16 & -1 \end{bmatrix} \mathbf{x}(kT) + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u(kT) \\ \mathbf{y}(kT) = [1 \quad 0] \mathbf{x}(kT) \end{cases}$$

2. 由 Z 传递函数建立离散状态空间表达式

一个线性离散系统可以用 Z 传递函数来表征, 当系统的 Z 传递函数已知时, 便可以建立该系统的离散状态空间表达式。常用方法有直接程序法、分式展开法、迭代程序法和嵌套程序法。

(1) 直接程序法

设线性离散系统的 Z 传递函数

$$G_c(z) = \frac{b_1 z^{-1} + b_2 z^{-2} + \cdots + b_n z^{-n}}{1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_n z^{-n}} \quad (m = n - 1)$$

$G_c(z)$ 是有理分式, 系统零点和极点不易求出, 可以采用直接程序法求离散状态空间表达式。令

$$\begin{aligned} Q(z) &= Y(z) / (b_1 z^{-1} + b_2 z^{-2} + \cdots + b_n z^{-n}) \\ &= U(z) / (1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_n z^{-n}) \end{aligned} \quad (6.25)$$

由式(6.25)可得

$$Q(z) = U(z) - a_1 z^{-1} Q(z) - a_2 z^{-2} Q(z) - \cdots - a_n z^{-n} Q(z) \quad (6.26)$$

$$Y(z) = b_1 z^{-1} Q(z) + b_2 z^{-2} Q(z) + \cdots + b_n z^{-n} Q(z) \quad (6.27)$$

由式(6.26), 式(6.27)可画出系统状态变量图, 如图6.3所示。

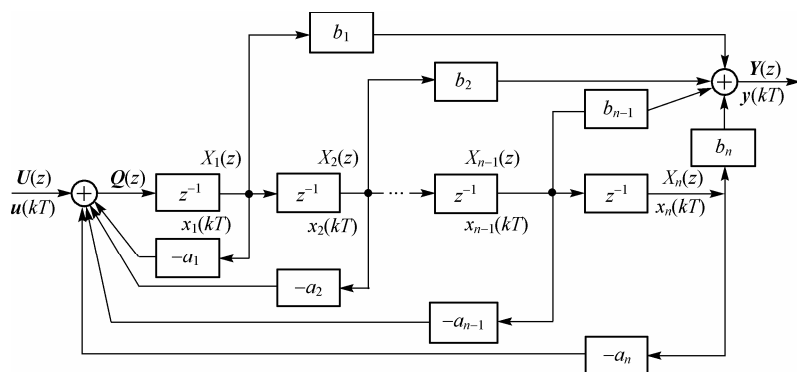


图 6.3 直接程序法建立的状态变量图

由图6.3可选择状态变量

$$\begin{cases} x_1(z) = z^{-1}Q(z) \\ x_2(z) = z^{-2}Q(z) = z^{-1}x_1(z) \\ \vdots \\ x_{n-1}(z) = z^{-n+1}Q(z) = z^{-1}x_{n-2}(z) \\ x_n(z) = z^{-n}Q(z) = z^{-1}x_{n-1}(z) \end{cases} \quad (6.28)$$

对式(6.28)进行 Z 逆变换

$$\begin{cases} x_2(kT+T) = x_1(kT) \\ x_3(kT+T) = x_2(kT) \\ \vdots \\ x_{n-1}(kT+T) = x_{n-2}(kT) \\ x_n(kT+T) = x_{n-1}(kT) \end{cases} \quad (6.29)$$

由式(6.26)及式(6.28), 可得

$$\begin{aligned} x_1(kT+T) = & -a_1x_1(kT) - a_2x_2(kT) - a_3x_3(kT) - \cdots \\ & - a_{n-1}x_{n-1}(kT) - a_nx_n(kT) + u(kT) \end{aligned} \quad (6.30)$$

由式(6.27), 可得输出方程

$$y(kT) = b_1x_1(kT) + b_2x_2(kT) + \cdots + b_{n-1}x_{n-1}(kT) + b_nx_n(kT) \quad (6.31)$$

由式(6.29), 式(6.30), 式(6.31)可得状态空间表达式

$$\begin{cases} \mathbf{x}(kT+T) = \mathbf{F}\mathbf{x}(kT) + \mathbf{G}u(kT) \\ \mathbf{y}(kT) = \mathbf{C}\mathbf{x}(kT) + \mathbf{D}u(kT) \end{cases} \quad (6.32)$$

$$\begin{aligned} \mathbf{F} &= \begin{bmatrix} -a_1 & -a_2 & \cdots & -a_{n-1} & -a_n \\ 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix} & \mathbf{G} &= \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \\ \mathbf{C} &= [b_1 \quad b_2 \quad \cdots \quad b_{n-1} \quad b_n] & \mathbf{D} &= [0] \end{aligned}$$

(2) 分式展开法

设线性离散系统的 Z 传递函数

$$\mathbf{G}_c(z) = \frac{\mathbf{Y}(z)}{\mathbf{U}(z)} = \frac{b_0z^m + b_1z^{m-1} + b_2z^{m-2} + \cdots + b_m}{(z+p_1)(z+p_2)\cdots(z+p_n)} \quad (m \leq n) \quad (6.33)$$

当 $\mathbf{G}_c(z)$ 的极点已经知道时, 可采用分式展开法, 下面分两种情况讨论。

① $\mathbf{G}_c(z)$ 具有不同极点 p_i 。

$$\mathbf{G}_c(z) = b_0 + \frac{d_1}{z+p_1} + \frac{d_2}{z+p_2} + \cdots + \frac{d_n}{z+p_n} \quad (6.34)$$

$$d_i = \lim_{z \rightarrow -p_i} [(z+p_i)\mathbf{G}_c(z)] \quad i=1,2,\cdots,n \quad (6.35)$$

$$Y(z) = \left(b_0 + \sum_{i=0}^n \frac{d_i}{z + p_i} \right) U(z) \quad (6.36)$$

由式(6.36)可得系统的状态变量图,如图6.4所示。

由图6.4选择状态变量,可得

$$X_i(z) = \frac{1}{z + p_i} U(z) \quad (6.37)$$

求 Z 逆变换,可得

$$x_i(kT + T) = -p_i x_i(kT) + u(kT) \quad (6.38)$$

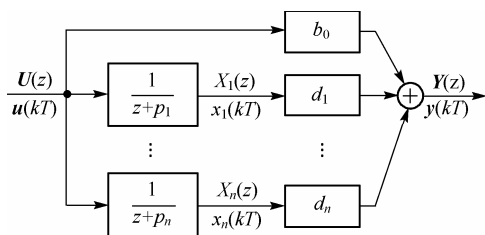


图 6.4 分式展开法相异极点时的状态变量图

由式(6.38), 式(6.37)及式(6.36)可得系统的离散状态空间表达式为

$$\begin{cases} \begin{bmatrix} x_1(kT + T) \\ x_2(kT + T) \\ \vdots \\ x_{n-1}(kT + T) \\ x_n(kT + T) \end{bmatrix} = \begin{bmatrix} -p_1 & 0 & \cdots & 0 & 0 \\ 0 & -p_2 & \cdots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \cdots & -p_{n-1} & 0 \\ 0 & 0 & \cdots & 0 & -p_n \end{bmatrix} \begin{bmatrix} x_1(kT) \\ x_2(kT) \\ \vdots \\ x_{n-1}(kT) \\ x_n(kT) \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \\ 1 \end{bmatrix} u(kT) \\ \\ y(kT) = [d_1 \quad d_2 \quad \cdots \quad d_{n-1} \quad d_n] \begin{bmatrix} x_1(kT) \\ x_2(kT) \\ \vdots \\ x_{n-1}(kT) \\ x_n(kT) \end{bmatrix} + [b_0] u(kT) \end{cases} \quad (6.39)$$

【例 6.3】 设线性离散系统的 Z 传递函数

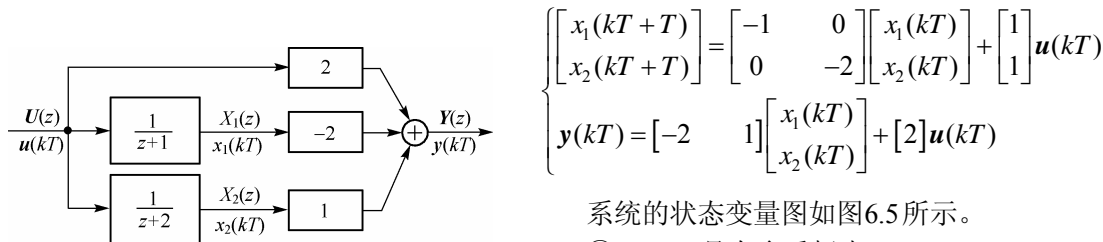
$$G_c(z) = \frac{2z^2 + 5z + 1}{(z+1)(z+2)}$$

试求系统的离散状态空间表达式。

解:

$$\begin{aligned} G_c(z) &= 2 + \frac{d_1}{z+1} + \frac{d_2}{z+2}, \quad b_0 = 2, p_1 = 1, p_2 = 2 \\ d_i &= \lim_{z \rightarrow -p_i} (z + p_i) G_c(z) \quad \text{得 } d_1 = -2, d_2 = 1 \end{aligned}$$

由式(6.39)可得系统的离散状态空间表达式



系统的状态变量图如图6.5所示。

② $G_c(z)$ 具有多重极点 p_n 。

设 $G_c(z) = -p_n$ 处有 l 重极点, 则

图 6.5 系统的状态变量图

$$G_c(z) = \frac{b_0 z^m + b_1 z^{m-1} + \cdots + b_{m-1} z + b_m}{(z + p_1)(z + p_2) \cdots (z + p_{n-l})(z + p_n)^l} \quad (6.40)$$

式中, $m \leq n$, 当 $m = n$ 时与前述相同, 可分部展开为

$$G_c(z) = b_0 + \frac{d_1}{z + p_1} + \frac{d_2}{z + p_2} + \cdots + \frac{d_{n-l}}{z + p_{n-l}} + \frac{e_l}{z + p_n} + \frac{e_{l-1}}{(z + p_n)^2} + \cdots + \frac{e_1}{(z + p_n)^l} \quad (6.41)$$

$$d_i = \lim_{z \rightarrow -p_i} [(z + p_i)G_c(z)] \quad i = 1, 2, \cdots, n-l \quad (6.42)$$

$$e_j = \frac{1}{(j-1)!} \lim_{z \rightarrow -p_n} \frac{d^{j-1}}{dz^{j-1}} [G_c(z)(z + p_n)^l] \quad j = l, l-1, \cdots, 1 \quad (6.43)$$

由式(6.41)可得系统状态变量图, 如图6.6所示。

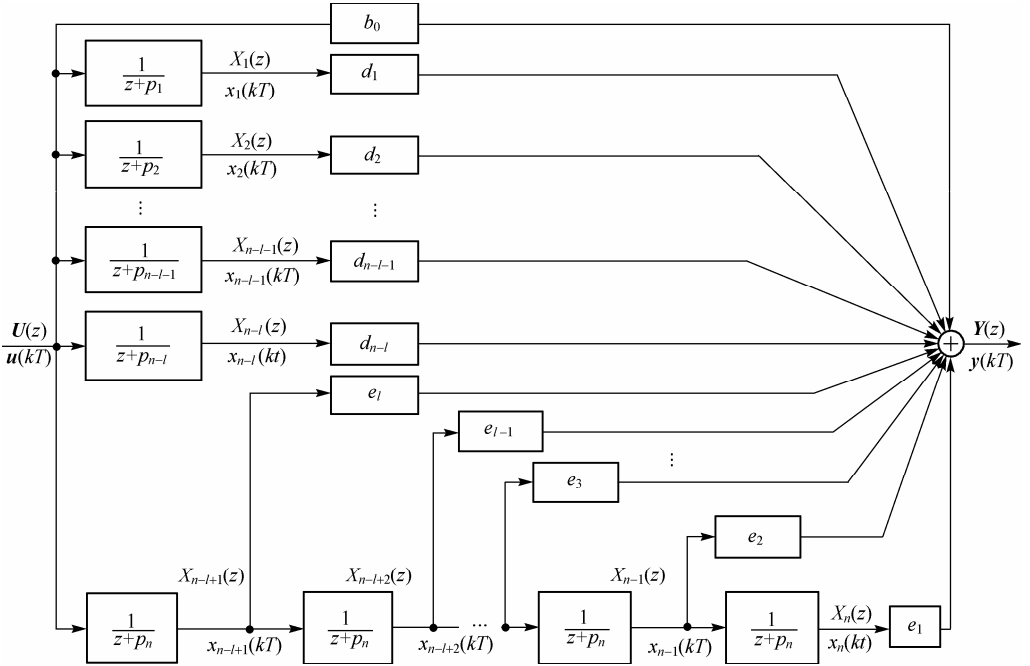


图 6.6 分式展开法系统有重极点时状态变量图

选择状态变量

$$X_i(z) = \frac{1}{z + p_i} U(z), \quad i = 1, 2, \cdots, n-l \quad (6.44)$$

$$X_{n-l+j}(z) = \frac{1}{(z + p_n)^j} U(z), \quad j = 1, 2, \cdots, l \quad (6.45)$$

对式(6.44)求 Z 逆变换, 即可得

$$x_i(kT + T) = -p_i x_i(kT) + u(kT) \quad i = 1, 2, \cdots, n-l \quad (6.46)$$

对式(6.45)求 Z 逆变换较为复杂

$$\begin{aligned} j=1 \quad & x_{n-l+1}(kT + T) = -p_n x_{n-l+1}(kT) + u(kT) \\ j=2 \quad & x_{n-l+2}(kT + T) = -p_n x_{n-l+2}(kT) + x_{n-l+1}(kT) \\ & \dots \end{aligned} \quad (6.47)$$

其余类推

$$x_{n-l+j}(kT+T) = -p_n x_{n-l+j}(kT) + x_{n-l+j-1}(kT)$$

由式(6.46), 式(6.47)及式(6.41)可推导出系统的离散状态空间表达式

$$\begin{cases} \mathbf{x}(kT+T) = \mathbf{F}\mathbf{x}(kT) + \mathbf{G}u(kT) \\ \mathbf{y}(kT) = \mathbf{C}\mathbf{x}(kT) + \mathbf{D}u(kT) \end{cases} \quad (6.48)$$

$$\mathbf{F} = \begin{bmatrix} -p_1 & 0 & 0 & \cdots & \cdots & \cdots & \cdots & 0 & 0 \\ 0 & -p_2 & 0 & \cdots & \cdots & \cdots & \cdots & 0 & 0 \\ \vdots & \vdots & & & & & & \vdots & \vdots \\ 0 & 0 & \cdots & -p_{n-l} & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \cdots & 0 & -p_n & 0 & \cdots & \cdots & 0 \\ \vdots & \vdots & \cdots & 0 & 1 & -p_n & \cdots & \cdots & 0 \\ \vdots & \vdots & \cdots & & & & & \vdots & \vdots \\ \vdots & \vdots & \cdots & \cdots & \cdots & \cdots & 1 & -p_n & 0 \\ 0 & 0 & \cdots & \cdots & \cdots & \cdots & 0 & 1 & -p_n \end{bmatrix} \quad (6.49)$$

$$\mathbf{G} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad \begin{array}{c} \updownarrow \\ n-l+1 \\ \updownarrow \\ l-1 \end{array} \quad (6.50)$$

$$\mathbf{C} = [d_1 \quad d_2 \quad \cdots \quad d_{n-l} \quad e_l \quad e_{l-1} \quad \cdots \quad e_1] \quad (6.51)$$

$$\mathbf{D} = [b_0] \quad (6.52)$$

当 $m < n$, 即 Z 传递函数 $G_c(z)$ 的分子的阶次低于分母的阶次时, $\mathbf{D} = [0]$

由式(6.49)可知状态矩阵 \mathbf{F} 是约当 (Jordan) 标准形。

(3) 迭代程序法

设线性离散系统的 Z 传递函数:

$$G_c(z) = \frac{b_0(z + \hat{z}_1)(z + \hat{z}_2) \cdots (z + \hat{z}_m)}{(z + p_1)(z + p_2) \cdots (z + p_n)} \quad (m \leq n) \quad (6.53)$$

当系统 Z 传递函数的零点和极点已知时, 可用迭代程序法求其离散状态空间表达式。

当 $m = n$ 时,

$$G_c(z) = b_0 + \frac{b(z + z_1)(z + z_2) \cdots (z + z_m)}{(z + p_1)(z + p_2) \cdots (z + p_n)} = b_0 + \frac{b}{z + p_1} \frac{z + z_1}{z + p_2} \cdots \frac{z + z_{n-1}}{z + p_n} \quad (6.54)$$

当 $m < n$ 时, 式(6.54)中 $b_0 = 0$ 。

由式(6.54)可得系统的状态变量图如图6.7所示。

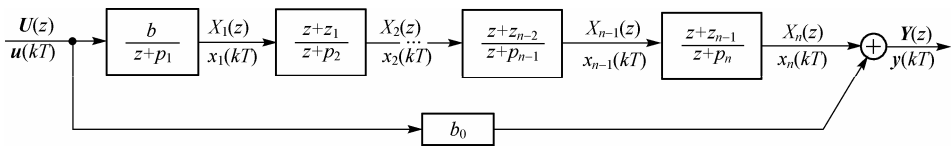


图 6.7 迭代程序法的状态变量图

由图6.7, 可列出状态空间表达式:

$$\begin{cases} x_1(kT+T) = -p_1x_1(kT) + bu(kT) \\ x_2(kT+T) = -p_2x_2(kT) + x_1(kT+T) + z_1x_1(kT) \\ \quad = (z_1 - p_1)x_1(kT) - p_2x_2(kT) + bu(kT) \\ x_3(kT+T) = (z_1 - p_1)x_1(kT) + (z_2 - p_2)x_2(kT) - p_3x_3(kT) + bu(kT) \\ \quad \vdots \\ x_n(kT+T) = (z_1 - p_1)x_1(kT) + (z_2 - p_2)x_2(kT) + \cdots \\ \quad + (z_{n-1} - p_{n-1})x_{n-1}(kT) - p_nx_n(kT) + bu(kT) \\ y(kT) = x_n(kT) + b_0u(kT) \end{cases} \quad (6.55)$$

由式(6.55)可得

$$\mathbf{F} = \begin{bmatrix} -p_1 & 0 & 0 & \cdots & 0 & 0 \\ (z_1 - p_1) & -p_2 & 0 & \cdots & 0 & 0 \\ (z_1 - p_1) & (z_2 - p_2) & -p_3 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ (z_1 - p_1) & (z_2 - p_2) & (z_3 - p_3) & \cdots & p_{n-1} & 0 \\ (z_1 - p_1) & (z_2 - p_2) & (z_3 - p_3) & \cdots & (z_{n-1} - p_{n-1}) & -p_n \end{bmatrix} \quad (6.56)$$

$$\mathbf{G} = \begin{bmatrix} b \\ b \\ b \\ \vdots \\ b \\ b \end{bmatrix} \quad (6.57)$$

$$\mathbf{C} = [0 \quad 0 \quad 0 \quad \cdots \quad 0 \quad 1] \quad (6.58)$$

$$\mathbf{D} = [b_0] \quad (6.59)$$

【例 6.4】设线性离散系统的 Z 传递函数

$$G_c(z) = \frac{2z^2 + 5z + 1}{z^2 + 3z + 2}$$

试求系统的离散状态空间表达式。

解:

$$G_c(z) = 2 + \frac{1}{(z+1)} \cdot \frac{-z-3}{z+2}$$

系统状态变量图如图6.8所示。

由状态变量, 可得

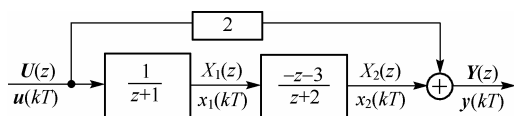


图 6.8 系统的状态变量图

$$\begin{cases} X_1(z) = \frac{1}{z+1}U(z) \\ X_2(z) = \frac{-z-3}{z+2}X_1(z) \\ Y(z) = X_2(z) + 2U(z) \end{cases} \quad (6.60)$$

由式(6.60), 可得

$$\begin{cases} x_1(kT+T) = -x_1(kT) + u(kT) \\ x_2(kT+T) = -x_1(kT+T) - 3x_1(kT) - 2x_2(kT) \\ \quad = -2x_1(kT) - 2x_2(kT) - u(kT) \\ y(kT) = x_2(kT) + 2u(kT) \end{cases} \quad (6.61)$$

由式(6.61), 可得离散状态空间表达式

$$\begin{cases} \begin{bmatrix} x_1(kT+T) \\ x_2(kT+T) \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ -2 & -2 \end{bmatrix} \begin{bmatrix} x_1(kT) \\ x_2(kT) \end{bmatrix} + \begin{bmatrix} 1 \\ -1 \end{bmatrix} u(kT) \\ y(kT) = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(kT) \\ x_2(kT) \end{bmatrix} + \begin{bmatrix} 2 \end{bmatrix} u(kT) \end{cases} \quad (6.62)$$

(4) 嵌套程序法

本方法与直接程序法一样, 能够适用于 $G_c(z)$ 的零极点未知的情况。

设系统的 Z 传递函数

$$G_c(z) = \frac{Y(z)}{U(z)} = \frac{b_0 z^n + b_1 z^{n-1} + \cdots + b_n}{z^n + a_1 z^{n-1} + \cdots + a_n} \quad (6.63)$$

或

$$G_c(z) = \frac{Y(z)}{U(z)} = \frac{b_0 + b_1 z^{-1} + \cdots + b_n z^{-n}}{1 + a_1 z^{-1} + \cdots + a_n z^{-n}} \quad (6.64)$$

由式(6.64)可得

$$Y(z) + a_1 z^{-1}Y(z) + \cdots + a_n z^{-n}Y(z) = b_0 U(z) + b_1 z^{-1}U(z) + \cdots + b_n z^{-n}U(z)$$

$$\begin{aligned} Y(z) &= b_0 U(z) + z^{-1}[b_1 U(z) - a_1 Y(z)] + z^{-2}[b_2 U(z) - a_2 Y(z)] \\ &\quad + \cdots + z^{-n}[b_n U(z) - a_n Y(z)] \end{aligned} \quad (6.65)$$

写成嵌套形式, 得

$$\begin{aligned} Y(z) &= b_0 U(z) + z^{-1}\{b_1 U(z) - a_1 Y(z) + z^{-1}[b_2 U(z) - a_2 Y(z)] \\ &\quad + z^{-1}(b_3 U(z) - a_3 Y(z) + \cdots)\} \end{aligned} \quad (6.66)$$

选择状态变量如下：

$$\begin{cases} X_n(z) = z^{-1}[b_n U(z) - a_n Y(z)] \\ X_{n-1}(z) = z^{-1}[b_{n-1} U(z) - a_{n-1} Y(z) + X_n(z)] \\ \vdots \\ X_2(z) = z^{-1}[b_2 U(z) - a_2 Y(z) + X_3(z)] \\ X_1(z) = z^{-1}[b_1 U(z) - a_1 Y(z) + X_2(z)] \end{cases} \quad (6.67)$$

由式(6.65)，得

$$Y(z) = b_0 U(z) + X_1(z) \quad (6.68)$$

将式(6.68)代入式(6.67)，并进行 Z 逆变换，可得差分方程组

$$\begin{cases} x_n(kT+T) = -a_n x_1(kT) + (b_n - a_n b_0)u(kT) \\ x_{n-1}(kT+T) = -a_{n-1} x_1(kT) + x_n(kT) + (b_{n-1} - a_{n-1} b_0)u(kT) \\ \vdots \\ x_2(kT+T) = -a_2 x_1(kT) + x_3(kT) + (b_2 - a_2 b_0)u(kT) \\ x_1(kT+T) = -a_1 x_1(kT) + x_2(kT) + (b_1 - a_1 b_0)u(kT) \end{cases} \quad (6.69)$$

$$y(kT) = x_1(kT) + b_0 u(kT) \quad (6.70)$$

由式(6.69)和式(6.70)可得状态变量图，如图6.9所示。

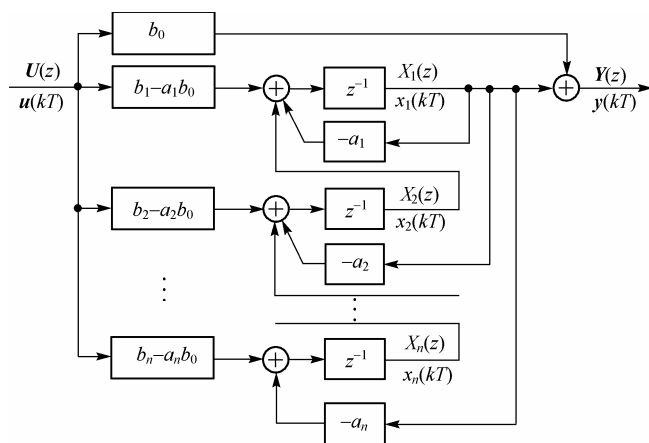


图 6.9 嵌套程序法的状态变量图

式(6.69)，式(6.70)的矩阵形式为

$$\begin{bmatrix} x_1(kT+T) \\ x_2(kT+T) \\ \vdots \\ x_{n-1}(kT+T) \\ x_n(kT+T) \end{bmatrix} = \begin{bmatrix} -a_1 & 1 & 0 & \cdots & 0 \\ -a_2 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -a_{n-1} & 0 & 0 & \cdots & 1 \\ -a_n & 0 & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} x_1(kT) \\ x_2(kT) \\ \vdots \\ x_{n-1}(kT) \\ x_n(kT) \end{bmatrix} + \begin{bmatrix} b_1 - a_1 b_0 \\ b_2 - a_2 b_0 \\ \vdots \\ b_{n-1} - a_{n-1} b_0 \\ b_n - a_n b_0 \end{bmatrix} u(kT) \quad (6.71)$$

$$\mathbf{y}(kT) = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1(kT) \\ x_2(kT) \\ \vdots \\ x_{n-1}(kT) \\ x_n(kT) \end{bmatrix} + b_0 \mathbf{u}(kT) \quad (6.72)$$

状态变量的选择不是唯一的, 对于同一个线性离散系统, 用不同的方法可以得到不同的离散状态空间表达式。但是状态变量的个数是相同的, 而且状态变量的个数跟系统的阶数相同。

6.1.2 线性离散系统状态方程的求解

线性离散系统离散状态方程是由高阶差分方程化为一阶差分方程组得到的, 所以求解差分方程的方法可以适用于求解离散状态方程。通常, 离散状态方程的求解方法有迭代法和 Z 变换法。

1. 迭代法

设线性离散系统的离散状态空间表达式为

$$\begin{cases} \mathbf{x}(kT + T) = \mathbf{F}\mathbf{x}(kT) + \mathbf{G}\mathbf{u}(kT) \\ \mathbf{y}(kT) = \mathbf{C}\mathbf{x}(kT) + \mathbf{D}\mathbf{u}(kT) \end{cases} \quad (6.73)$$

初始值 $\mathbf{x}(0), \mathbf{u}(0)$ 。

将 $k=0, 1, 2, \cdots$, 代入式(6.73), 可得

$$\begin{aligned} \mathbf{x}(T) &= \mathbf{F}\mathbf{x}(0) + \mathbf{G}\mathbf{u}(0) \\ \mathbf{x}(2T) &= \mathbf{F}\mathbf{x}(T) + \mathbf{G}\mathbf{u}(T) = \mathbf{F}^2\mathbf{x}(0) + \mathbf{F}\mathbf{G}\mathbf{u}(0) + \mathbf{G}\mathbf{u}(T) \\ \mathbf{x}(3T) &= \mathbf{F}^3\mathbf{x}(0) + \mathbf{F}^2\mathbf{G}\mathbf{u}(0) + \mathbf{F}\mathbf{G}\mathbf{u}(T) + \mathbf{G}\mathbf{u}(2T) \\ &\vdots \\ \mathbf{x}(kT) &= \mathbf{F}^k\mathbf{x}(0) + \sum_{j=0}^{k-1} \mathbf{F}^{k-j-1}\mathbf{G}\mathbf{u}(jT) \end{aligned} \quad (6.74)$$

根据离散状态方程的解式(6.74), 可以得到输出方程

$$\mathbf{y}(kT) = \mathbf{C}\mathbf{x}(kT) + \mathbf{D}\mathbf{u}(kT) \quad (6.75)$$

式(6.74)中, 记 $\Phi(kT) = \mathbf{F}^k$, $\Phi(kT)$ 称为离散系统的转移矩阵。

【例 6.5】 用迭代法求解线性离散系统状态方程

$$\begin{cases} \mathbf{x}(kT + T) = \begin{bmatrix} 0 & -1 \\ -0.4 & 0.3 \end{bmatrix} \mathbf{x}(kT) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \mathbf{u}(kT) \\ \mathbf{y}(kT) = \begin{bmatrix} 0 & 1 \end{bmatrix} \mathbf{x}(kT) \\ \mathbf{x}(0) = \begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \mathbf{u}(kT) = \begin{cases} 1 & k \geq 0 \\ 0 & k < 0 \end{cases} \end{cases}$$

解: 令 $k=0, 1, 2, \cdots$, 及初始条件代入离散状态空间表达式, 可以得到

$$\mathbf{x}(T) = \begin{bmatrix} 0 & -1 \\ -0.4 & 0.3 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} [1] = \begin{bmatrix} 1 \\ 0.3 \end{bmatrix}$$

$$y(T) = [0 \quad 1] \begin{bmatrix} 1 \\ 0.3 \end{bmatrix} = 0.3$$

$$\mathbf{x}(2T) = \begin{bmatrix} 0 & -1 \\ -0.4 & 0.3 \end{bmatrix} \begin{bmatrix} 1 \\ 0.3 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} [1] = \begin{bmatrix} -0.3 \\ 0.69 \end{bmatrix}$$

$$y(2T) = [0 \quad 1] \begin{bmatrix} -0.3 \\ 0.69 \end{bmatrix} = 0.69$$

$$\mathbf{x}(3T) = \begin{bmatrix} 0 & -1 \\ -0.4 & 0.3 \end{bmatrix} \begin{bmatrix} -0.3 \\ 0.69 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} [1] = \begin{bmatrix} -0.69 \\ 1.327 \end{bmatrix}$$

$$y(3T) = [0 \quad 1] \begin{bmatrix} -0.69 \\ 1.327 \end{bmatrix} = 1.327$$

⋮

k	0	1	2	3	...
$x_1(kT)$	1	1	-0.3	-0.69	...
$x_2(kT)$	-1	0.3	0.69	1.327	...
$y(kT)$	-1	0.3	0.69	1.327	...

用迭代法求解离散状态方程只能得到有限项时间序列，得不到状态变量和输出变量的数学解析式。

2. Z 变换法

设线性离散系统的状态空间表达式为

$$\begin{cases} \mathbf{x}(kT + T) = \mathbf{F}\mathbf{x}(kT) + \mathbf{G}\mathbf{u}(kT) \\ \mathbf{y}(kT) = \mathbf{C}\mathbf{x}(kT) + \mathbf{D}\mathbf{u}(kT) \end{cases} \quad (6.76)$$

对式(6.76)进行 Z 变换，可得

$$\begin{cases} z\mathbf{X}(z) - z\mathbf{x}(0) = \mathbf{F}\mathbf{X}(z) + \mathbf{G}\mathbf{U}(z) \\ \mathbf{X}(z) = (z\mathbf{I} - \mathbf{F})^{-1}[z\mathbf{x}(0) + \mathbf{G}\mathbf{U}(z)] \end{cases} \quad (6.77)$$

对式(6.77)进行 Z 逆变换，可得

$$\mathbf{x}(kT) = Z^{-1}\{(z\mathbf{I} - \mathbf{F})^{-1}[z\mathbf{x}(0) + \mathbf{G}\mathbf{U}(z)]\} \quad (6.78)$$

对比式(6.74)，下式成立

$$\Phi(kT) = Z^{-1}[(z\mathbf{I} - \mathbf{F})^{-1}z] \quad (6.79)$$

【例 6.6】用 Z 变换法求解线性离散状态方程

$$\mathbf{x}(kT + T) = \begin{bmatrix} 0 & -1 \\ -0.4 & 0.3 \end{bmatrix} \mathbf{x}(kT) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \mathbf{u}(kT)$$

$$\mathbf{y}(kT) = [0 \quad 1] \mathbf{x}(kT)$$

$$\mathbf{x}(0) = \begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \mathbf{u}(kT) = \begin{cases} 1 & k \geq 0 \\ 0 & k < 0 \end{cases}$$

解:

$$\mathbf{x}(kT) = Z^{-1} \{ (z\mathbf{I} - \mathbf{F})^{-1} [z\mathbf{x}(0) + \mathbf{G}U(z)] \}$$

$$(z\mathbf{I} - \mathbf{F})^{-1} = \begin{bmatrix} z & 1 \\ 0.4 & z - 0.3 \end{bmatrix}^{-1} = \begin{bmatrix} \frac{z - 0.3}{(z - 0.8)(z + 0.5)} & \frac{-1}{(z - 0.8)(z + 0.5)} \\ \frac{-0.4}{(z - 0.8)(z + 0.5)} & \frac{z}{(z - 0.8)(z + 0.5)} \end{bmatrix}$$

$$z\mathbf{x}(0) + \mathbf{G}U(z) = z \begin{bmatrix} 1 \\ -1 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} z \\ z - 1 \end{bmatrix} = \begin{bmatrix} z \\ \frac{2z - z^2}{z - 1} \end{bmatrix}$$

$$\begin{aligned} (z\mathbf{I} - \mathbf{F})^{-1} [z\mathbf{x}(0) + \mathbf{G}U(z)] &= \begin{bmatrix} \frac{z - 0.3}{(z - 0.8)(z + 0.5)} & \frac{-1}{(z - 0.8)(z + 0.5)} \\ \frac{-0.4}{(z - 0.8)(z + 0.5)} & \frac{z}{(z - 0.8)(z + 0.5)} \end{bmatrix} \begin{bmatrix} z \\ \frac{2z - z^2}{z - 1} \end{bmatrix} \\ &= \begin{bmatrix} \frac{z(z^2 - 0.3z - 1.7)}{(z - 0.8)(z + 0.5)(z - 1)} \\ \frac{-z(z^2 - 1.6z - 0.4)}{(z - 0.8)(z + 0.5)(z - 1)} \end{bmatrix} = \begin{bmatrix} \frac{5z}{z - 0.8} - \frac{2z}{3(z + 0.5)} - \frac{5z}{3(z - 1)} \\ \frac{-4z}{z - 0.8} - \frac{z}{3(z + 0.5)} + \frac{10z}{3(z - 1)} \end{bmatrix} \end{aligned}$$

取 Z 逆变换, 得到方程的解

$$\begin{aligned} \mathbf{x}(kT) &= \begin{bmatrix} 5(0.8)^k - 2(-0.5)^k / 3 - 10/3 \\ -4(0.8)^k - (-0.5)^k / 3 + 10/3 \end{bmatrix} \\ \mathbf{y}(kT) &= \begin{bmatrix} -4(0.8)^k - (-0.5)^k / 3 + 10/3 \end{bmatrix} \end{aligned}$$

用 Z 变换法求解离散状态方程, 可以得到状态变量和输出变量的数学解析式。

6.1.3 线性离散系统的 Z 传递矩阵和 Z 特征方程

1. 线性离散系统的 Z 传递矩阵

设线性离散系统的状态空间表达式为

$$\begin{cases} \mathbf{x}(kT + T) = \mathbf{F}\mathbf{x}(kT) + \mathbf{G}u(kT) \\ \mathbf{y}(kT) = \mathbf{C}\mathbf{x}(kT) + \mathbf{D}u(kT) \end{cases} \quad (6.80)$$

式中, $\mathbf{x}(kT)$ 是 $n \times 1$ 维状态向量; $u(kT)$ 是 $m \times 1$ 维输入向量; $\mathbf{y}(kT)$ 是 $p \times 1$ 维输出向量。

对式(6.80)进行 Z 变换, 可得

$$\begin{cases} z\mathbf{X}(z) - z\mathbf{x}(0) = \mathbf{F}\mathbf{X}(z) + \mathbf{G}U(z) \\ \mathbf{Y}(z) = \mathbf{C}\mathbf{X}(z) + \mathbf{D}U(z) \end{cases}$$

当初始条件为零, 即 $\mathbf{x}(0) = 0$ 时

$$\mathbf{Y}(z) = [\mathbf{C}(z\mathbf{I} - \mathbf{F})^{-1}\mathbf{G} + \mathbf{D}]\mathbf{U}(z) = \mathbf{G}_c(z)\mathbf{U}(z)$$

$$\mathbf{G}_c(z) = [\mathbf{C}(z\mathbf{I} - \mathbf{F})^{-1}\mathbf{G} + \mathbf{D}] \quad (6.81)$$

称 $\mathbf{G}_c(z)$ 为线性离散系统的 Z 传递矩阵 ($p \times m$ 维矩阵)。它反映了在初始静止的条件下, 输出量的 Z 变换 $\mathbf{y}(z)$ 与输入量的 Z 变换 $\mathbf{U}(z)$ 之间的关系。

对于单输入单输出系统, $G_c(z)$ 是 1×1 维矩阵, 即为 Z 传递函数 $G_c(z)$ 。

【例 6.7】 设线性离散系统的状态空间表达式为

$$\begin{cases} \mathbf{x}(kT+T) = \begin{bmatrix} 0 & -1 \\ -0.4 & 0.3 \end{bmatrix} \mathbf{x}(kT) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(kT) \\ \mathbf{y}(kT) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \mathbf{x}(kT) \end{cases}$$

初始条件为零。试求线性离散系统的 Z 传递矩阵, 并求出单位阶跃输入时的输出响应。

解:

$$\begin{aligned} (z\mathbf{I} - \mathbf{F})^{-1} &= \begin{bmatrix} \frac{z-0.3}{(z-0.8)(z+0.5)} & \frac{-1}{(z-0.8)(z+0.5)} \\ \frac{-0.4}{(z-0.8)(z+0.5)} & \frac{z}{(z-0.8)(z+0.5)} \end{bmatrix} \\ G_c(z) = [C(z\mathbf{I} - \mathbf{F})^{-1}\mathbf{G} + D] &= \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{z-0.3}{(z-0.8)(z+0.5)} & \frac{-1}{(z-0.8)(z+0.5)} \\ \frac{-0.4}{(z-0.8)(z+0.5)} & \frac{z}{(z-0.8)(z+0.5)} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} \frac{z-0.7}{(z-0.8)(z+0.5)} & \frac{z-1}{(z-0.8)(z+0.5)} \\ \frac{-0.4}{(z-0.8)(z+0.5)} & \frac{z}{(z-0.8)(z+0.5)} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{z-1}{(z-0.8)(z+0.5)} \\ \frac{z}{(z-0.8)(z+0.5)} \end{bmatrix} \end{aligned}$$

单位阶跃输入时,

$$U(z) = \frac{z}{z-1}$$

$$\begin{aligned} Y(z) = G_c(z)U(z) &= \begin{bmatrix} \frac{z-1}{(z-0.8)(z+0.5)} \\ \frac{z}{(z-0.8)(z+0.5)} \end{bmatrix} \frac{z}{z-1} = \begin{bmatrix} \frac{z}{(z-0.8)(z+0.5)} \\ \frac{z^2}{(z-1)(z-0.8)(z+0.5)} \end{bmatrix} \\ &= \begin{bmatrix} \frac{10z}{13(z-0.8)} - \frac{10z}{13(z+0.5)} \\ \frac{13z}{3.9(z-1)} - \frac{12z}{3.9(z-0.8)} - \frac{z}{3.9(z+0.5)} \end{bmatrix} \end{aligned}$$

对上式进行 Z 逆变换, 有

$$y(kT) = \begin{bmatrix} (10/13)(0.8)^k - (10/13)(-0.5)^k \\ (10/3) - (40/13)(0.8)^k - (10/39)(-0.5)^k \end{bmatrix}$$

2. 线性离散系统的 Z 特征方程

在线性连续系统中, 用特征方程来表征系统的动态特性, 同样, 引进 Z 特征方程的概念来描述一个线性离散系统的动态特性。

设线性离散系统的状态方程为

$$\mathbf{x}(kT+T) = \mathbf{F}\mathbf{x}(kT) + \mathbf{G}u(kT) \quad (6.82)$$

对式(6.82)进行 Z 变换, 可得

$$\mathbf{X}(z) = (z\mathbf{I} - \mathbf{F})^{-1} [z\mathbf{x}(0) + \mathbf{G}U(z)] \quad (6.83)$$

仿照线性连续系统, 令矩阵 $(z\mathbf{I} - \mathbf{F})$ 的行列式为

$$|z\mathbf{I} - \mathbf{F}| = 0 \quad (6.84)$$

则称式(6.84)为线性离散系统的 Z 特征方程。

【例 6.8】 设线性离散系统的状态矩阵

$$\mathbf{F} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

试求线性离散系统的 Z 特征方程。

解:

$$\begin{aligned} |z\mathbf{I} - \mathbf{F}| &= \begin{vmatrix} z & 0 \\ 0 & z \end{vmatrix} - \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = \begin{vmatrix} z - a_{11} & -a_{12} \\ -a_{21} & z - a_{22} \end{vmatrix} \\ &= z^2 - (a_{11} + a_{22})z + a_{11}a_{22} - a_{12}a_{21} \end{aligned}$$

$$\text{令} \quad z^2 - (a_{11} + a_{22})z + a_{11}a_{22} - a_{12}a_{21} = 0 \quad (6.85)$$

式(6.85)为线性离散系统的 Z 特征方程或称为矩阵 \mathbf{F} 的 Z 特征方程。 Z 特征方程的根也称为矩阵 \mathbf{F} 的特征值, 就是线性离散系统的极点。

对于一个 n 阶的系统, 仅有 n 个特征值。因为特征方程表征了系统的动态特性, 因此尽管一个系统的状态变量的选择不是唯一的, 但是系统的 Z 特征方程是不变的。

【例 6.9】 设线性离散系统的 Z 传递函数为

$$\mathbf{G}_c(z) = \frac{2z^2 + 5z + 1}{z^2 + 3z + 2}$$

要求导出离散状态方程, 并求出 Z 特征方程。

解: 用直接程序法导出离散状态方程, 且

$$\mathbf{F} = \begin{bmatrix} -3 & -2 \\ 1 & 0 \end{bmatrix}$$

$$Z \text{ 特征方程为} \quad z^2 + 3z + 2 = 0$$

$$\text{特征值为} \quad p_1 = -2, p_2 = -1$$

当然, 也可以用分式展开法和嵌套法导出同样的结果。

6.1.4 微型计算机控制系统的离散状态空间表达式

微型计算机控制系统的离散状态空间表达式, 可以通过连续环节的状态空间表达式离散化得到, 也可以如例 6.10 用 Z 变换得到。

【例 6.10】已知计算机控制系统如图6.10所示，初始静止。

试求系统的离散状态空间表达式。

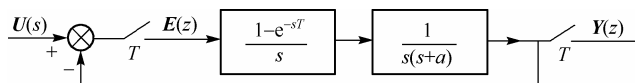


图 6.10 典型的计算机控制系统

解：广义（带零阶保持器）对象的传递函数

$$HG(s) = \frac{1 - e^{-sT}}{s^2(s+a)}$$

广义对象 z 传递函数

$$HG(z) = \frac{T}{a(z-1)} - \frac{1 - e^{-aT}}{a^2(z - e^{-aT})} \quad (6.86)$$

由式(6.86)可以建立系统的方框图，如图6.11所示。

选择状态变量

$$\begin{cases} X_1(z) = \frac{T}{a(z-1)}[U(z) - X_1(z) - X_2(z)] \\ X_2(z) = \frac{e^{-aT} - 1}{a^2(z - e^{-aT})}[U(z) - X_1(z) - X_2(z)] \end{cases} \quad (6.87)$$

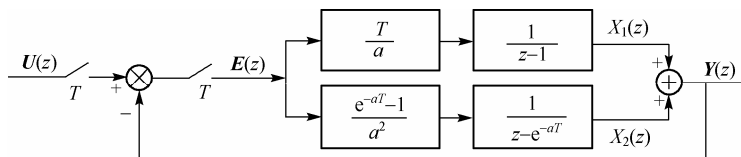


图 6.11 系统方框等效图

由式(6.87)可得

$$\begin{cases} zX_1(z) = \left(1 - \frac{T}{a}\right)X_1(z) - \frac{T}{a}X_2(z) + \frac{T}{a}U(z) \\ zX_2(z) = \frac{1 - e^{-aT}}{a^2}X_1(z) + \frac{a^2e^{-aT} - e^{-aT} + 1}{a^2}X_2(z) + \frac{e^{-aT} - 1}{a^2}U(z) \end{cases} \quad (6.88)$$

由式(6.88)可得离散状态方程，由图6.11可得离散状态空间表达式

$$\begin{cases} \begin{bmatrix} x_1(kT+T) \\ x_2(kT+T) \end{bmatrix} = \begin{bmatrix} 1 - \frac{T}{a} & -\frac{T}{a} \\ \frac{1 - e^{-aT}}{a^2} & \frac{(a^2 - 1)e^{-aT} + 1}{a^2} \end{bmatrix} \begin{bmatrix} x_1(kT) \\ x_2(kT) \end{bmatrix} + \begin{bmatrix} \frac{T}{a} \\ \frac{e^{-aT} - 1}{a^2} \end{bmatrix} u(kT) \\ y(kT) = \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} x_1(kT) \\ x_2(kT) \end{bmatrix} \end{cases} \quad (6.89)$$

当 $a=1$ 时,

$$\mathbf{F} = \begin{bmatrix} 1-T & -T \\ 1-e^{-T} & 1 \end{bmatrix}$$

系统 Z 特征方程为

$$|z\mathbf{I} - \mathbf{F}| = \begin{vmatrix} z-1+T & T \\ e^{-T}-1 & z-1 \end{vmatrix} = z^2 + (T-2)z + (1-Te^{-T}) = 0$$

将连续部分的状态空间表达式离散化, 也即要求导出连续部分的离散状态方程, 假设微机控制系统的方框图如图6.12所示。

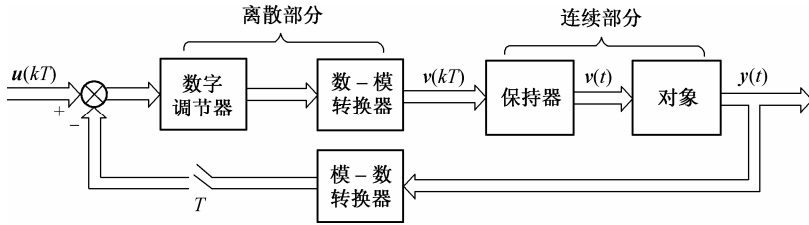


图 6.12 微机控制系统方框图

由图 6.12 可见, 微机控制系统由连续部分和离散部分组成。连续部分就是保持器和控制对象。绝大多数情况下, 使用的是零阶保持器, 其输出 $v(t)$ 是阶梯信号。

连续对象的动态特性可以用如下状态空间表达式表示:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{v}(t) \\ \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{v}(t) \\ \mathbf{x}(t_0) = \mathbf{x}(0) \end{cases} \quad (6.90)$$

式(6.90)的解为

$$\mathbf{x}(t) = e^{\mathbf{A}(t-t_0)}\mathbf{x}(0) + \int_{t_0}^t e^{\mathbf{A}(t-\tau)}\mathbf{B}\mathbf{v}(\tau)d\tau \quad (6.91)$$

当 $v(t)$ 是阶梯信号时,

$$\mathbf{v}(t) = \mathbf{v}(kT) = \text{常数}, \quad kT \leq t < kT + T \quad (6.92)$$

初始条件为

$$\mathbf{x}(0) = \mathbf{x}(t_0) = \mathbf{x}(kT)$$

积分限为

$$\begin{aligned} t &= kT + T \\ t_0 &= kT, \text{ 且 } t - t_0 = T \end{aligned} \quad (6.93)$$

由式(6.91), 可得

$$\mathbf{x}(kT + T) = e^{\mathbf{A}T}\mathbf{x}(kT) + \int_{kT}^{kT+T} e^{\mathbf{A}[kT+T-t]}\mathbf{B}\mathbf{v}(kT)d\tau \quad (6.94)$$

在积分区间内, 输入是常数, 而且积分对所有的 k 都成立, 进行变量置换: $t = kT + T - \tau$, 则有

$$\int_{kT}^{kT+T} e^{A[kT+T-t]} B dt = \int_0^T e^{At} B dt \quad (6.95)$$

将式(6.95)代入式(6.94)可得

$$\mathbf{x}(kT+T) = e^{AT} \mathbf{x}(kT) + \left(\int_0^T e^{At} B dt \right) \mathbf{v}(kT) \quad (6.96)$$

即为所要求的离散状态方程，写成标准形式时为

$$\begin{cases} \mathbf{x}(kT+T) = \mathbf{F}\mathbf{x}(kT) + \mathbf{G}\mathbf{v}(kT) \\ \mathbf{y}(kT) = \mathbf{C}\mathbf{x}(kT) + \mathbf{D}\mathbf{v}(kT) \end{cases} \quad (6.97)$$

式中

$$\begin{aligned} \mathbf{F} &= e^{AT} \\ \mathbf{G} &= \int_0^T e^{At} B dt \end{aligned}$$

6.2 离散系统的能控性和能观测性

由 Kalman 提出的控制系统的能控性和能观测性的概念，在多变量最优控制系统中具有重要意义。事实上，能控性和能观测性可以给出最优控制问题存在完整解的条件。

能控性指的是控制作用对被控系统影响的可能性。如果在一个有限的时间间隔里，可以用一个无约束的控制向量，使得系统由初始状态 $\mathbf{x}(t_0)$ 转移到重点状态 $\mathbf{x}(t_f)$ ，那么系统就称为在时间 t_0 是能控的。

能观测性反映了由系统的量测，确定系统状态的可能性。如果系统在状态 $\mathbf{x}(t_0)$ ，可通过在一个有限的时间间隔内，由输出量的观测值确定，那么系统就称为在时间 t_0 是能观测的。

能控性和能观测性从状态的控制能力和状态的测辨能力两个方面揭示了控制系统构成的两个基本问题。

尽管大多数物理系统是能控和能观测的，然而，也有部分物理系统可能不具有能控性和能观测性。如果所研究的系统是不能控的，那么，最优控制问题的解不存在。因此，应当清楚系统在什么条件下是能控的或能观测的。

6.2.1 离散系统的能控性

设离散系统的离散状态方程为

$$\mathbf{x}(kT+T) = \mathbf{F}\mathbf{x}(kT) + \mathbf{G}\mathbf{u}(kT) \quad (6.98)$$

式中， $\mathbf{x}(kT)$ 为状态向量 n 维； $\mathbf{u}(kT)$ 为控制向量 m 维； \mathbf{F} 为状态矩阵， $n \times n$ 维，是非奇异矩阵； \mathbf{G} 为输入矩阵， $n \times m$ 维； T 为采样周期。

为了讨论方便，假设 $\mathbf{u}(kT)$ 在 $kT \leq t \leq kT+T$ 的时间内是一个常值。如果在有限采样间隔 $0 \leq kT \leq NT$ 内，存在阶梯信号 $\mathbf{u}(kT)$ ，驱动系统由任意初始状态开始，经过 NT 进入状态 $\mathbf{x}(kT)$ ，那么由式(6.98)所确定的离散系统是能控的。如果每一个状态都是能控的，那么系统被称为是状态完全能控的。

方程(6.98)的解为

$$\mathbf{x}(kT) = \mathbf{F}^k \mathbf{x}(0) + \sum_{j=0}^{k-1} \mathbf{F}^j \mathbf{G} u(kT - jT - T) \quad (6.99)$$

NT 时刻的解为

$$\mathbf{x}(NT) = \mathbf{F}^N \mathbf{x}(0) + \sum_{j=0}^{N-1} \mathbf{F}^j \mathbf{G} u(NT - jT - T) \quad (6.100)$$

或者

$$\mathbf{x}(NT) - \mathbf{F}^N \mathbf{x}(0) = \mathbf{F}^{N-1} \mathbf{G} u(0) + \mathbf{F}^{N-2} \mathbf{G} u(T) + \cdots + \mathbf{G} u(NT - T) \quad (6.101)$$

引入 n 维向量

$$\mathbf{h}_k = \mathbf{F}^k \mathbf{G}, \quad k = 0, 1, 2, \dots, N-1 \quad (6.102)$$

则式(6.101)可表示为

$$\mathbf{x}(NT) - \mathbf{F}^N \mathbf{x}(0) = \mathbf{h}_0 \mathbf{u}(NT - T) + \mathbf{h}_1 \mathbf{u}(NT - 2T) + \cdots + \mathbf{h}_{N-1} \mathbf{u}(0) \quad (6.103)$$

初始状态 $\mathbf{x}(0)$ 及终止状态 $\mathbf{x}(NT)$ 为任意给定值时, 为了满足式(6.103), 在向量组 $\{\mathbf{h}_0, \mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_{N-1}\}$ 中必须有 n 个独立的向量, 这是因为控制变量 $u(0), u(T), u(2T), \dots, u(NT - T)$ 都为标量。在 $\mathbf{x}(0)$, $\mathbf{x}(NT)$ 任意给定时, 达到 $\mathbf{x}(NT)$ 所需的步数 N 不可能低于系统的阶数 n , 也就是 $N \geq n$ 。设 $N = n$, 用线性方程表示式(6.103), 可有

$$\begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{h}_0 & \mathbf{h}_1 & \cdots & \mathbf{h}_{n-1} & \begin{bmatrix} u(nT - T) \\ u(nT - 2T) \\ \vdots \\ u(0) \end{bmatrix} \end{bmatrix} = [\mathbf{x}(nT) - \mathbf{F}^n \mathbf{x}(0)] \quad (6.104)$$

为了使控制序列 $u(0), u(T), u(2T), \dots, u(NT - T)$ 在式(6.104)右边取任何值时都能够存在, 系数矩阵的各个列向量必须线性独立, 也就是满足

$$[\mathbf{h}_0 \quad \mathbf{h}_1 \quad \cdots \quad \mathbf{h}_{n-1}] = \text{rank}[\mathbf{G}, \mathbf{F}\mathbf{G}, \mathbf{F}^2\mathbf{G}, \dots, \mathbf{F}^{N-1}\mathbf{G}] = n \quad (\text{系统的阶数}) \quad (6.105)$$

如果式(6.105)成立, 那么就能够在有限拍时间内使系统的状态从 $\mathbf{x}(0)$ 转移到 $\mathbf{x}(NT)$, 就称系统是完全能控的。应当注意, 以上结论只有当 $u(kT)$ 不受约束时, 才是正确的。如果 $u(kT)$ 受约束, 那么必须大于 n 个采样周期。

设 $N = n$, 式(6.101)的两边左乘 $-\mathbf{F}^{-n}$ (\mathbf{F} 必须正则), 则

$$\mathbf{x}(0) - \mathbf{F}^{-n} \mathbf{x}(nT) = -\mathbf{F}^{-1} \mathbf{G} u(0) - \mathbf{F}^{-2} \mathbf{G} u(T) - \cdots - \mathbf{F}^{-n} \mathbf{G} u(nT - T) \quad (6.106)$$

令

$$\mathbf{f}_k = -\mathbf{F}^{-k} \mathbf{G}, \quad k = 1, 2, \dots, n \quad (6.107)$$

则式(6.106)可表示为

$$\mathbf{x}(0) - \mathbf{F}^{-n} \mathbf{x}(nT) = \mathbf{f}_1 \mathbf{u}(0) + \mathbf{f}_2 \mathbf{u}(T) + \cdots + \mathbf{f}_n \mathbf{u}(nT - T) \quad (6.108)$$

因此, 状态完全能控的条件也可以表达为式(6.107)表示的 \mathbf{f}_n 是线性独立的。

如果控制向量 $\mathbf{u}(kT)$ 是 m 维向量, 则系统的能控性仍旧由上述相同的条件决定。

【例 6.11】设二阶系统如图6.13所示，采样周期 $T=1\text{ s}$ ，试判断系统的能控性。

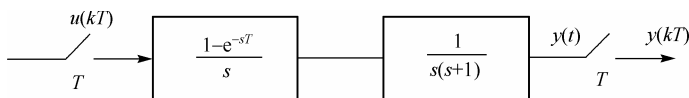


图 6.13 二阶系统的能控性

解：二阶系统的离散状态空间表达式为

$$\left. \begin{aligned} \begin{bmatrix} x_1(kT+T) \\ x_2(kT+T) \end{bmatrix} &= \begin{bmatrix} 0.368 & 0 \\ 0.632 & 1 \end{bmatrix} \begin{bmatrix} x_1(kT) \\ x_2(kT) \end{bmatrix} + \begin{bmatrix} 0.632 \\ 0.368 \end{bmatrix} u(kT) \\ y(kT) &= x_2(kT) \end{aligned} \right\} \quad (6.109)$$

所以

$$\left. \begin{aligned} F^{-1} &= \begin{bmatrix} 0.368 & 0 \\ 0.632 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 2.718 & 0 \\ -1.718 & 1 \end{bmatrix} \\ G &= \begin{bmatrix} 0.632 \\ 0.368 \end{bmatrix} \end{aligned} \right\} \quad (6.110)$$

由式(6.110)可得

$$\left. \begin{aligned} f_1 &= -F^{-1}G = -\begin{bmatrix} 2.718 & 0 \\ -1.718 & 1 \end{bmatrix} \begin{bmatrix} 0.632 \\ 0.368 \end{bmatrix} = \begin{bmatrix} -1.718 \\ 0.718 \end{bmatrix} \\ f_2 &= -F^{-2}G = -\begin{bmatrix} 2.718 & 0 \\ -1.718 & 1 \end{bmatrix}^2 \begin{bmatrix} 0.632 \\ 0.368 \end{bmatrix} = \begin{bmatrix} -4.669 \\ 3.669 \end{bmatrix} \end{aligned} \right\} \quad (6.111)$$

因此

$$\text{rank}[f_1, f_2] = \text{rank} \begin{bmatrix} -1.718 & -4.669 \\ 0.718 & 3.669 \end{bmatrix} = 2 \quad (6.112)$$

在控制向量不受约束的情况下，系统是完全能控的。

如果离散系统是由 Z 传递函数描述，则该离散系统能控的条件是 Z 传递函数的分子和分母不存在对消因子，否则离散系统是不能控的。

【例 6.12】设离散系统的 Z 传递函数为

$$G_c(z) = \frac{Y(z)}{U(z)} = \frac{z+a}{(z+a)(z+b)} \quad (6.113)$$

试判断系统的能控性。

解：由系统的 Z 传递函数可得

$$\begin{aligned} (z+a)(z+b)Y(z) &= (z+a)U(z) \\ [z^2 + (a+b)z + ab]Y(z) &= (z+a)U(z) \end{aligned} \quad (6.114)$$

系统的差分方程为

$$y(kT+2T) + (a+b)y(kT+T) + aby(kT) = u(kT+T) + au(kT) \quad (6.115)$$

设 $x_1(kT) = y(kT)$ ， $x_2(kT) = x_1(kT+T) - u(kT)$

所以

$$\left. \begin{aligned} x_1(kT+T) &= x_2(kT) + u(kT) \\ x_2(kT+T) &= -abx_1(kT) - (a+b)x_2(kT) - bu(kT) \end{aligned} \right\} \quad (6.116)$$

离散系统的离散状态方程为

$$\begin{bmatrix} \mathbf{x}_1(kT+T) \\ \mathbf{x}_2(kT+T) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -ab & -(a+b) \end{bmatrix} \begin{bmatrix} \mathbf{x}_1(kT) \\ \mathbf{x}_2(kT) \end{bmatrix} + \begin{bmatrix} 1 \\ -b \end{bmatrix} \mathbf{u}(kT) \quad (6.117)$$

离散系统的能控性矩阵

$$[\mathbf{GFG}] = \left\{ \begin{bmatrix} 1 \\ -b \end{bmatrix} \begin{bmatrix} 0 & 1 \\ -ab & -(a+b) \end{bmatrix} \begin{bmatrix} 1 \\ -b \end{bmatrix} \right\} = \begin{bmatrix} 1 & -b \\ -b & b^2 \end{bmatrix} = 0 \quad (6.118)$$

由式(6.118)可知, 能控性矩阵是奇异的, 所以离散系统是不能控的。

能控性反映了系统的状态向量从初始状态转移到所希望的状态的可能性。同样, 能否使输出向量 $\mathbf{y}(kT)$ 转移到所希望的数值也是一个很重要的问题, 由于输出向量和状态向量之间存在如下的关系:

$$\mathbf{y}(kT) = \mathbf{C}\mathbf{x}(kT) \quad (6.119)$$

所以, 由式(6.105)可以证明, 输出的能控性条件是

$$\text{rank}[\mathbf{CG}, \mathbf{CFG}, \mathbf{CF}^2\mathbf{G}, \dots, \mathbf{CF}^{N-1}\mathbf{G}] = p \quad (6.120)$$

式中, p 是输出向量的维数。

6.2.2 离散系统的能观测性

系统极点配置的时候需要全状态变量反馈, 但是能否测量和重构全部状态, 就要判断系统的能观测性, 也就是在有限的步数内 (与初始状态无关), 分析测量和重构所有状态的可能性, 能观测性取决于系统的特性矩阵 \mathbf{F} 和 \mathbf{C} 。

设离散系统的状态空间表达式为

$$\begin{cases} \mathbf{x}(kT+T) = \mathbf{F}\mathbf{x}(kT) + \mathbf{G}\mathbf{u}(kT) \\ \mathbf{y}(kT) = \mathbf{C}\mathbf{x}(kT) \end{cases} \quad (6.121)$$

式中, $\mathbf{y}(kT)$ 是 p 维输出向量; $\mathbf{x}(kT)$ 是 n 维状态向量; $\mathbf{u}(kT)$ 是 m 维控制向量; \mathbf{F} 是 $n \times n$ 维状态矩阵; \mathbf{G} 是 $n \times m$ 维输入矩阵; \mathbf{C} 是 $p \times n$ 维输出矩阵。

如果给出有限个采集周期内的输出 $\mathbf{y}(kT)$, 就可以确定系统的初始状态向量 $\mathbf{x}(0)$, 那么系统是可观测的。

设从 0 瞬间开始测量 n 次 (每隔周期 T 测量一次), 因为

$$\begin{cases} \mathbf{x}(kT) = \mathbf{F}^k \mathbf{x}(0) \\ \mathbf{y}(kT) = \mathbf{C}\mathbf{F}^k \mathbf{x}(0) \end{cases} \quad (6.122)$$

式(6.122)中不包含由 $\mathbf{u}(kT)$ 引起的分量, 因为 \mathbf{F} , \mathbf{G} , \mathbf{C} 和 $\mathbf{u}(kT)$ 都已知时, 该分量可从 $\mathbf{x}(kT)$, $\mathbf{y}(kT)$ 中扣除。由 n 次测量可得

$$\begin{cases} \mathbf{y}(0) = \mathbf{C}\mathbf{x}(0) \\ \mathbf{y}(T) = \mathbf{C}\mathbf{F}\mathbf{x}(0) \\ \vdots \\ \mathbf{y}(nT-T) = \mathbf{C}\mathbf{F}^{n-1}\mathbf{x}(0) \end{cases} \quad (6.123)$$

式(6.123)的矩阵形式为

$$\begin{bmatrix} \mathbf{y}(0) \\ \mathbf{y}(T) \\ \vdots \\ \mathbf{y}(nT-T) \end{bmatrix} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CF} \\ \vdots \\ \mathbf{CF}^{n-1} \end{bmatrix} \mathbf{x}(0) \quad (6.124)$$

写成初始值 $\mathbf{x}(0)$ 的表达式

$$\mathbf{x}(0) = \begin{bmatrix} \mathbf{C} \\ \mathbf{CF} \\ \vdots \\ \mathbf{CF}^{n-1} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{y}(0) \\ \mathbf{y}(T) \\ \vdots \\ \mathbf{y}(nT-T) \end{bmatrix} \quad (6.125)$$

离散系统完全能观测是根据 $\mathbf{y}(0), \mathbf{y}(T), \dots, \mathbf{y}(nT-T)$ ，由式(6.125)可以确定 $x_1(0), x_2(0), \dots, x_n(0)$ 。为了确定 n 个未知数，需要 $\mathbf{y}(kT)$ 的 n 组值，因此时间序列从 0 到 $nT-T$ 。为了求得 $x_1(0), x_2(0), \dots, x_n(0)$ 的一组唯一解，矩阵

$$\begin{bmatrix} \mathbf{C} \\ \mathbf{CF} \\ \vdots \\ \mathbf{CF}^{n-1} \end{bmatrix} \quad (6.126)$$

中应该找出 n 个线性无关的方程，也就是式(6.126)矩阵的秩应为 n ，也即

$$\text{rank} \begin{bmatrix} \mathbf{C} \\ \mathbf{CF} \\ \vdots \\ \mathbf{CF}^{n-1} \end{bmatrix} = n \quad (6.127)$$

【例 6.13】 设系统如图 6.13 所示，试分析系统的能观测性。

解：由式(6.109)知

$$\mathbf{F} = \begin{bmatrix} 0.368 & 0 \\ 0.632 & 1 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 0 & 1 \end{bmatrix}$$

所以

$$\text{rank} \begin{bmatrix} \mathbf{C} \\ \vdots \\ \mathbf{CF} \end{bmatrix} = \text{rank} \begin{bmatrix} \begin{bmatrix} 0 & 1 \end{bmatrix} \\ \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} 0.368 & 0 \\ 0.632 & 1 \end{bmatrix} \end{bmatrix} = \text{rank} \begin{bmatrix} 0 & 1 \\ 0.632 & 1 \end{bmatrix} = 2 \quad (6.128)$$

离散系统是完全能观测的。

【例 6.14】 设系统的运动方程为

$$\ddot{\theta} = \frac{M}{I} \quad (6.129)$$

试讨论系统的能观测性。

解：设 $\theta_1 = \theta$ ， $\theta_2 = \dot{\theta}$ ，系统的状态方程为

$$\begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \frac{M}{I} \quad (6.130)$$

离散化以后, 得

$$\begin{bmatrix} \theta_1(kT+T) \\ \theta_2(kT+T) \end{bmatrix} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \theta_1(kT) \\ \theta_2(kT) \end{bmatrix} + \begin{bmatrix} 0 \\ T \end{bmatrix} \frac{M}{I} \quad (6.131)$$

式中, T 是采样周期。如果 θ 是可测量的, 则

$$C = \begin{bmatrix} 1 & 0 \end{bmatrix} \quad (6.132)$$

离散系统的能观测矩阵为

$$\begin{bmatrix} C \\ CF \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & T \end{bmatrix} \quad (6.133)$$

式(6.133)矩阵是非奇异的, 所以系统是能观测的。

如果 $\dot{\theta}$ 是可以测量的, θ 是不能测量的, 则 $C = \begin{bmatrix} 1 & 0 \end{bmatrix}$, 系统的能观测矩阵为

$$\begin{bmatrix} C \\ CF \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} = 0 \quad (6.134)$$

式(6.134)是奇异的, 所以系统是不能观测的。

由上所述, 如果 θ 能够测量, 则在有限的步数内 (决定于精度的要求) 可观测到 $\dot{\theta}$ 。如果仅有 $\dot{\theta}$ 可以测量, 则状态 θ 是不能观测的, 因为必须知道初始值和 $\dot{\theta}$ 的时间函数才能求得 θ 值。

6.3 离散系统的状态空间设计

6.3.1 MIMO线性离散系统的状态空间设计方法

设多输入-多输出系统如图6.14所示, 离散状态空间设计法的目标是利用离散状态空间表达式, 设计出数字调节器 $D(z)$, 使得微型计算机控制系统满足或者达到要求的性能指标。微机控制系统中, 控制对象经常是连续对象, 对象的特性用状态空间表达式表示为

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) \end{cases} \quad (6.135)$$

式中, $\mathbf{x}(t)$ 是 n 维状态向量; $\mathbf{u}(t)$ 是 m 维控制向量; $\mathbf{y}(t)$ 是 p 维输出向量; \mathbf{A} 是 $n \times n$ 维状态矩阵; \mathbf{B} 是 $n \times m$ 维输入矩阵; \mathbf{C} 是 $p \times n$ 维输出矩阵。

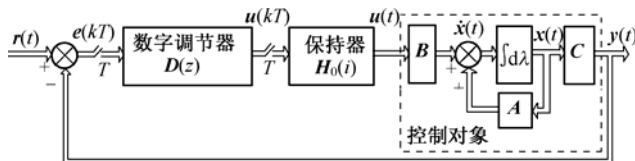


图 6.14 多输入-多输出系统的方框图

设计时应首先把对象离散化, 用离散状态空间表达式表征控制对象。

离散状态空间设计法的步骤如下:

① 对连续对象离散化, 设采样周期为 T , 对象的离散状态空间表达式为

$$\begin{cases} \mathbf{x}(kT+T) = \mathbf{F}\mathbf{x}(kT) + \mathbf{G}\mathbf{u}(kT) \\ \mathbf{y}(kT) = \mathbf{C}\mathbf{x}(kT) \end{cases} \quad (6.136)$$

式中, $\mathbf{x}(kT)$ 是 n 维状态向量; $\mathbf{u}(kT)$ 是 m 维控制向量; $\mathbf{y}(kT)$ 是 p 维输出向量; \mathbf{F} 是 $n \times n$ 维状态矩阵; \mathbf{G} 是 $n \times m$ 维输入矩阵; \mathbf{C} 是 $p \times n$ 维输出矩阵。

\mathbf{F} , \mathbf{G} 与式(6.135)中的 \mathbf{A} , \mathbf{B} 有关

$$\mathbf{F} = \mathbf{e}^{AT} \quad (6.137)$$

$$\mathbf{G} = \int_0^T \mathbf{e}^{AT} \mathbf{B} \, dt \quad (6.138)$$

式(6.136)和式(6.135)中的 \mathbf{C} 相同。

② 计算能够使 $\mathbf{y}(t)$ 经过 N 个采样周期单调地达到稳态的数字调节器的输出序列 $\mathbf{u}(kT)$ 。

③ 计算误差序列 $\mathbf{e}(kT)$

$$\mathbf{e}(kT) = \mathbf{r}(kT) - \mathbf{y}(kT) \quad (6.139)$$

④ 分别对 $\mathbf{u}(kT)$, $\mathbf{e}(kT)$ 取 Z 变换, 取两者之比, 即可求得数字调节器的 Z 传递矩阵 $\mathbf{D}(z)$

$$\mathbf{D}(z) = \frac{Z[\mathbf{u}(kT)]}{Z[\mathbf{e}(kT)]} \quad (6.140)$$

6.3.2 MIMO离散系统设计实例

周期激励下的结构振动是一种常见的现象, 如偏载的转子系统, 受发动机激励的火箭系统, 电动机因转矩变化对执行机构产生的激励, 液压系统流体的周期脉动对管路系统的影响等。这种由周期激励引起的持续周期振动对结构和控制系统是极其有害的, 必须加以抑制。

下面通过一个航天柔性结构周期振动的主被动控制实例来说明多输入-多输出系统的离散控制器设计方法。

随着航空航天技术的飞速发展, 对空间结构系统的性能提出了越来越高的要求, 大型化、低刚度和柔性化是各类航天结构的一个重要发展趋势: 大型结构可以增加空间结构的功能, 如大的太阳能电池阵列可以为空间结构提供更加充足的能源; 轻型结构则可以增加有效载荷的重量, 提高运载工具的效率。这些大型空间柔性结构 (Large Space Flexible Structure, LSFS), 如大型模块化的宇宙空间站、太阳能帆板、卫星天线、高精度光学系统及其支承结构、空间机器臂等, 需要能在相当长的运行时间内保证高的运行精度。但它们刚度低、内阻小、太空环境又无外阻, 如不采取措施对其进行振动抑制, 在太空中运行时一旦受到某种激励力的作用, 其大幅度的振动要持续很长时间。这不仅会直接影响航天结构的运行精度, 如持续的振动可妨碍太阳能帆板面跟踪太阳, 卫星天线和光学仪器的指向精度, 以及空间机械手的定位精度等, 还将使结构产生过早的疲劳破坏, 影响结构的使用寿命, 导致结构上仪器的损坏。因此, LSFS 中振动控制的研究是迫在眉睫的任务。

主被动一体化振动控制 (Integrated Active and Passive Vibration Control, IAPVC) 是目前航天柔性结构振动控制的一个主要发展趋势, 它将主动控制和被动控制结合, 进行优势互补, 以达到更好的效果。IAPVC 技术基本上可分为两类: 第一类是组合控制 (Combined Control), 即在一个结构中同时配置被动控制系统和主动控制系统, 对结构同时施加主动控制和被动控制; 第二类是杂交控制 (Hybrid Control), 将主被动控制方法统一, 构成一体化系统。这里的主被动一体化振动控制技术属于组合控制, 给出的航天 (仪器舱) 柔性结构振动控制系统如图 6.15 所示。

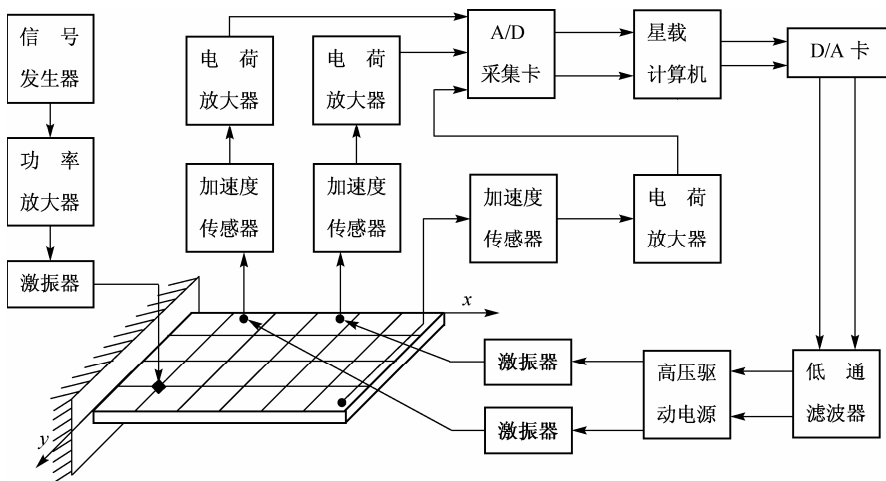


图 6.15 悬臂铝合金薄板振动控制系统框图

在将重复控制应用到柔性结构的周期振动控制的基础上，采用“零极点对消 + 零相位偏差跟踪”的方法设计了单输入单输出改进型离散时间重复控制器，采用“逆相对对消”的方法设计了多输入多输出改进型离散时间重复控制器；然后，设计了改进型多频离散时间重复控制器，以解决具有不同频率的多个信号激励系统的振动控制问题；最后，以悬臂铝合金薄板和悬臂约束阻尼薄板为研究对象，应用以上重复控制器，对两薄板的周期振动分别进行了主动和主被动控制。

1. 重复控制基本原理

重复控制（Repetitive Control, RC）是因质子同步加速器主环电源系统中的电磁铁电流控制需要，而由东京工业大学中野道雄教授等提出的控制系统设计理论，它被认为是一种能很好地解决实际工业过程控制问题的快速、高精度实用控制方法。近年来，重复控制理论得到了广泛的重视，在各个领域的应用研究中取得了显著成果。

重复控制原理是把周期为 T 的任意目标信号作为跟踪目标。周期为 T 的任意周期函数可以通过给出一个对应于一个周期的任意初始函数，并把它存储起来，每隔周期 T 就重复取出来而产生。因此，构造如图6.16所示的周期为 T 的周期函数发生器，它含有时间为 T 的时滞环节。假设时滞环节 e^{-Ts} 中的初始函数为 $f_0(x)$ ，则目标函数 $r(t)$ 可表示为

$$r(iT + x) = f_0(x), \quad 0 \leq x \leq T, i = 0, 1, 2, \dots \quad (6.141)$$

根据内部模型原理，只要把上述发生器作为内部模型放置在闭环内，并使闭环系统稳定，就可以构成对周期为 T 的任意目标信号均无稳态偏差的伺服系统。这个周期函数发生器称为重复补偿器，而设置了重复补偿器的控制系统称为重复控制系统，其基本构成如图6.17所示。

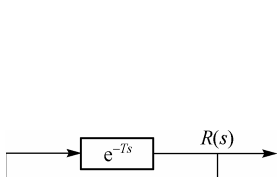


图 6.16 周期函数发生器

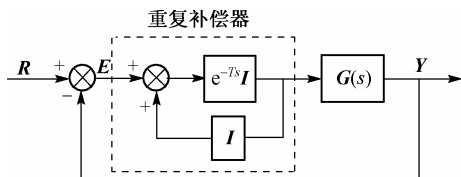


图 6.17 典型重复控制系统

系统中包含的内部模型为

$$H(s) = \frac{1}{e^{Ts} - 1} I \quad (6.142)$$

与连续时间系统的情形相对应，在离散系统中，周期函数发生器如图 6.18 所示，这里目标输入的周期 T 、采样周期 T_c 和滞后段数 N 之间的关系为 $T = T_c N$ 。同样，如果把这个周期函数发生器放置在闭环内，并使整个闭环系统稳定，则可以实现无稳态偏差地跟踪周期目标输入的离散时间重复控制系统（Discrete Time Repetitive Control System, DTRCS）。

值得注意的是，连续时间系统的内部模型为 $1/(e^{Ts}-1)$ ，是无限维的线性系统，包含该内部模型的连续时间重复控制系统也是无限维的，因而系统的稳定化很困难。而离散时间系统的内部模型为 $1/(z^N-1)$ ，是有限维（ N 维）的线性系统，包含该内部模型的离散时间重复控制系统也是有限维的，它的稳定化可以用通常的方法来实现。所以，采用离散时间重复控制系统的优点在于：①离散的方法在设计和实现上都比连续的方法简便；②可以避免连续系统在离散化过程中出现的不稳定情况，尤其是对于小阻尼系统。

这里离散时间重复控制系统的设计方法是“零极点对消+零相位偏差跟踪”及“逆相位对消”的方法，是一种改进型离散重复控制系统。

2. 零相位偏差跟踪前馈控制器

(1) 完全跟踪控制器

为了得到很好的跟踪性能，系统除了设计反馈控制器外，还需要设计前馈控制器，如图 6.19 所示。

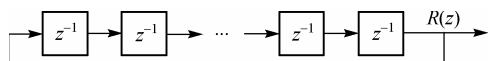


图 6.18 离散时间周期函数发生器

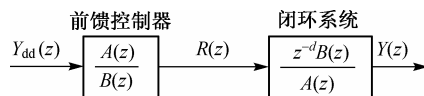


图 6.19 完全跟踪前馈控制器

设包含控制对象和反馈控制器的离散闭环传函为

$$P(z) = \frac{z^{-d} B(z)}{A(z)} \quad (6.143)$$

式中， d 为由控制对象引起的滞后步数。

$$\begin{aligned} B(z) &= b_0 + b_1 z^{-1} + \cdots + b_m z^{-m}, \quad b_0 \neq 0 \\ A(z) &= 1 + a_1 z^{-1} + \cdots + a_n z^{-n}, \quad n \geq m \end{aligned}$$

由图 6.19，有

$$Y(z) = P(z)R(z) \quad (6.144)$$

$$R(z) = \frac{A(z)}{B(z)} Y_{dd}(z) = \frac{z^d A(z)}{B(z)} Y_d(z) \quad (6.145)$$

式中， $Y_d(z)$ 为期望输出， $Y_{dd}(z)$ 为超前 d 步的期望输出。

如果 $A(z)=0$ 和 $B(z)=0$ 的根都在 z 平面以原点为圆心的单位圆中，即闭环传递函数的零极点都在单位圆中，则初始条件为零时，闭环传递函数和前馈控制器的所有零极点都可以对消，于是 $Y(z) = Y_d(z)$ ，称这种前馈控制器为完全跟踪控制器（Perfect Tracking Control, PTC）。

(2) 零相位偏差跟踪控制器

如果闭环传函零点，也就是 PTC 的极点不全在单位圆内，或者在单位圆内，但位于某些

不利的位置，如在负实轴上且接近-1，这时 PTC 的输出 $R(z)$ 会发散或激烈振荡。于是，将 $B(z)$ 分解为两部分

$$B(z) = B^a(z)B^u(z) \quad (6.146)$$

式中， $B^a(z)$ 包含可以对消的零点； $B^u(z)$ 包含不可以对消的零点，即那些在单位圆外、在单位圆上和在单位圆内处于不利位置的零点。

$$B^u(z) = b^u_0 + b^u_1 z^{-1} + \cdots + b^u_s z^{-s}, \quad b^u_0 \neq 0, s \leq m \quad (6.147)$$

则实现与所有闭环传函极点及 $B^a(z)$ 中零点对消的前馈控制器为

$$R(z) = \frac{A(z)}{B^a(z)B^u(1)} Y_{dd}^*(z) = \frac{z^d A(z)}{B^a(z)B^u(1)} Y_d^*(z) \quad (6.148)$$

式中， $Y_d^*(z)$ 与期望输出有关； $Y_{dd}^*(z)$ 为超前 d 步的 $Y_d^*(z)$ ； $B^u(1)$ 为令 $B^u(z)$ 中 $z=1$ 所得的常数。

设初始条件为零，由式(6.144)、式(6.146)和式(6.148)可得

$$Y(z) = \frac{z^{-d} B^u(z)}{B^u(1)} Y_{dd}^*(z) = \frac{B^u(z)}{B^u(1)} Y_d^*(z) \quad (6.149)$$

式(6.149)表明，如果期望输出能表示为

$$Y_d(z) = \frac{B^u(z)}{B^u(1)} Y_d^*(z) \quad (6.150)$$

则可以实现完全跟踪。

由于闭环传递函数存在不能对消的零点，式(6.148)所示的前馈控制器会产生跟踪偏差。为了消除相位偏差，这里引入

$$\left(\frac{B^u(z)}{B^u(1)} \right) \left(\frac{B^{u+}(z)}{B^u(1)} \right) = 1 \quad (6.151)$$

$$B^{u+}(z) = b^u_0 + b^u_1 z + \cdots + b^u_s z^s$$

于是，可以通过式(6.150)由 $Y_d(z)$ 产生 $Y_d^*(z)$ ，即

$$Y_d^*(z) = \left(\frac{B^{u+}(z)}{B^u(1)} \right) Y_d(z) \quad (6.152)$$

由式(6.148)和式(6.152)，得到前馈控制器为

$$R(z) = \frac{z^d A(z) B^{u+}(z)}{B^a(z) (B^u(1))^2} Y_d(z) = \frac{z^{d+s} A(z) B^{u*}(z)}{B^a(z) (B^u(1))^2} Y_d(z) \quad (6.153)$$

$$B^{u*}(z) = b^u_s + b^u_{(s-1)} z^{-1} + \cdots + b^u_0 z^{-s}$$

这时，对于所有频率， $Y_d(z)$ 和 $Y(z)$ 之间的相移为零，称该前馈控制器为零相位偏差跟踪控制器（Zero Phase Error Tracking Controller, ZPETC），如图6.20所示。图中， $Y_{dd}^*(z)$ 为超前 d 步的 $Y_d^*(z)$ ， $Y_{dds}(z)$ 为超前 $d+s$ 步的 $Y_d(z)$ 。

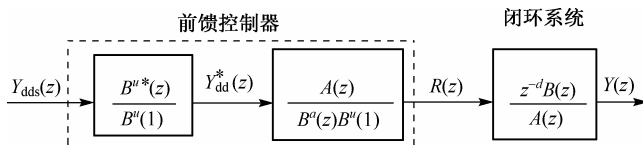


图 6.20 零相位偏差跟踪前馈控制器

3. MIMO 离散重复控制器的设计

(1) 系统描述

设控制对象的离散状态方程为

$$\begin{aligned}\mathbf{x}(k+1) &= \mathbf{F}_d \mathbf{x}(k) + \mathbf{G}_d \mathbf{u}(k) \\ \mathbf{y}(k) &= \mathbf{C}_d \mathbf{x}(k)\end{aligned}\quad (6.154)$$

式中, $\mathbf{x}(k)$ 为 p 维状态向量, $\mathbf{u}(k)$ 为 m 维输入向量, $\mathbf{y}(k)$ 为 n 维输出向量, \mathbf{F}_d 为 $p \times p$ 矩阵, \mathbf{G}_d 为 $p \times m$ 矩阵, \mathbf{C}_d 为 $n \times p$ 矩阵, 控制对象是能控的、稳定的, 且为最小相位系统。

于是, 控制对象的 z 域传递矩阵为

$$\mathbf{P}(z) = \mathbf{C}_d (\mathbf{I}_p - z^{-1} \mathbf{F}_d)^{-1} z^{-1} \mathbf{G}_d \quad (6.155)$$

MIMO 离散重复控制系统的框图如图6.21所示。

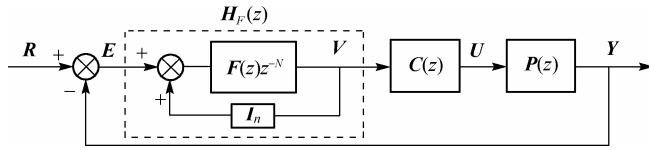


图 6.21 MIMO 改进型离散时间重复控制系统

系统的传递矩阵为

$$\mathbf{H}_{YR}(z) = \left(z^N \mathbf{I}_n - \mathbf{F}(z) + \mathbf{F}(z) \mathbf{P}(z) \mathbf{C}(z) \right)^{-1} \mathbf{F}(z) \mathbf{P}(z) \mathbf{C}(z) \quad (6.156)$$

式中, $\mathbf{P}(z)$ 为 $n \times m$ 控制对象传递矩阵; $\mathbf{C}(z)$ 为 $m \times n$ 待设计稳定化补偿器; $\mathbf{F}(z)$ 为 $n \times n$ 低通滤波器对角矩阵, 对角上的元素可按式(6.157)设计, $\mathbf{F}(z)$ 可独立设计而不影响系统的稳定性。

$$\mathbf{F}(z) = \frac{\sum_{i=0}^{l_F} a_i z^i + \sum_{i=1}^{l_F} a_i z^{-i}}{2 \sum_{i=1}^{l_F} a_i + a_0} \quad (6.157)$$

令传递矩阵 \mathbf{H}_{YR} 的分母等于零, 就得到了闭环系统的特征方程, 当且仅当该特征方程的根位于 z 平面中以原点为圆心的单位圆内时, 系统稳定。关于 MIMO 改进型重复控制系统的稳定性, 由下面定理确定:

定理 1 可以构成 MIMO 改进型重复控制系统的充要条件是 $\mathbf{P}(z)$ 与 $1 - z^{-N}$ 没有相同的零点, 即对于 $\omega \in [0, 2\pi/N, 4\pi/N, \dots, [N/2] \cdot 2\pi/N]$, $\text{rank}(\mathbf{P}(z) | z = e^{j\omega T_c}) = n$ 。

定理 2 对于 $\mathbf{F}(z) = \mathbf{I}_n$, 如果

$$\|\mathbf{I}_n - \mathbf{P}(z) \mathbf{C}(z)\| \leq 1 \quad (6.158)$$

成立, $\|\cdot\|$ 为范数, 则 MIMO 改进型重复控制系统渐进稳定。

考虑图 6.21 中输入 $\mathbf{R}(z)$ 与输出 $\mathbf{Y}(z)$ 维数相同, 由定理 2 可知, $\mathbf{C}(z)$ 设计的一种可能选择是

$$\mathbf{P}(z) \mathbf{C}(z) = \varepsilon_d \mathbf{I}_n, \quad 0 < \varepsilon_d < 2 \quad (6.159)$$

由于 $\mathbf{P}(z)$ 为方阵且其所有元素都是最小相位传递函数，所以当 $\varepsilon_d = 1$ 时， $\mathbf{C}(z)$ 很自然地取为 $\mathbf{P}(z)^{-1}$ 。

(2) 逆相位对消矩阵

由上面的分析，考虑矩阵 $\mathbf{P}(z)$ 的各元素为有理传递函数。令 $\Psi_d = \mathbf{P}(z)\mathbf{P}'(z)$ ， ψ_{dij} 为 Ψ_d 的第 (i, j) 元素，如果对于 $\omega \in [0, 2\pi/N, 4\pi/N, \dots, [N/2] \cdot 2\pi/N]$ ，有

$$\begin{cases} |\psi_{dij}(z)|_{i=j} = c_i \\ |\psi_{dij}(z)|_{i \neq j} = 0, c_i \text{ 为任意实数}, z = e^{j\omega} \\ \angle \psi_{dij}(z) = 0 \end{cases} \quad (6.160)$$

则称 $\mathbf{P}'(z)$ 为 $\mathbf{P}(z)$ 的逆相位对消矩阵 (Phase Cancellation Inverse Matrix, PCIM)。

可以看出，如果 $\mathbf{C}(z)$ 设计为逆相位对消矩阵，则 MIMO 改进型重复控制系统总是可以满足式(6.158)。

如果矩阵 $\mathbf{P}(z)$ 的各元素为有理传递函数，则其逆相位对消矩阵为

$$\begin{aligned} \det(\mathbf{P}(z)) &= \beta_p^a(z) \beta_p^u(z) \\ \mathbf{P}'(z) &= \text{Adj}(\mathbf{P}(z)) \frac{\beta_p^u(z)}{\beta_p^a(z)} \end{aligned} \quad (6.161)$$

式中， $\beta_p^a(z)$ 和 $\beta_p^u(z)$ 分别为 $\det(\mathbf{P}(z))$ 的可对消部分和不可对消部分。

于是，由式(6.155)所示控制对象传递矩阵，根据矩阵逆的定义，有

$$\begin{aligned} \mathbf{S}_p(z) &= \mathbf{C}_d \text{Adj}(\mathbf{I}_p - z^{-1} \mathbf{F}_d) z^{-1} \mathbf{G}_d \\ \mathbf{P}(z) &= \frac{\mathbf{S}_p(z)}{\det(\mathbf{I}_p - z^{-1} \mathbf{F}_d)} \end{aligned} \quad (6.162)$$

式中， $\text{Adj}(\cdot)$ 表示矩阵的伴随阵； $\det(\cdot)$ 表示矩阵的行列式。

令

$$\begin{aligned} \mathbf{T}_p(z) &= \text{Adj}(\mathbf{S}_p(z)) \\ \det(\mathbf{T}_p(z)) &= \beta_T^a(z) \beta_T^u(z) \end{aligned} \quad (6.163)$$

式中， $\beta_T^a(z)$ 和 $\beta_T^u(z)$ 分别为 $\det(\mathbf{T}_p(z))$ 的可对消部分和不可对消部分。

由定理 2，将 $\mathbf{C}(z)$ 设计为 $\mathbf{P}(z)$ 的逆相位对消矩阵为

$$\mathbf{C}(z) = \mathbf{P}'(z) = \det(\mathbf{I}_p - z^{-1} \mathbf{F}_d) \frac{\beta_T^u(z)}{\beta_T^a(z)} \mathbf{T}_p(z) \quad (6.164)$$

4. 悬臂约束阻尼薄板振动的主被动控制效果

对于悬臂约束阻尼薄板，控制对象是能控的、稳定的，且为最小相位系统，采样频率 $T_c = 3600 \text{ Hz}$ 。

对于薄板上点(0.1224, 0.1556)，低通滤波器 $F(z) = 1/4(2 + z + z^{-1})$ 。控制对象离散状态方程中各矩阵和 z 域传递函数为

$$\mathbf{F}_d = \begin{bmatrix} 0.9980 & 0 & 0 & 0 \\ 0 & 0.9826 & 0 & 0 \\ -14.0500 & 0 & 0.9786 & 0 \\ 0 & -124.3340 & 0 & 0.9407 \end{bmatrix}$$

$$\mathbf{G}_d = \begin{bmatrix} 0 \\ 0 \\ 0.0235 \\ -0.0094 \end{bmatrix} \quad \mathbf{C}_d = [0 \quad 0 \quad 5.7423 \quad -3.2083]$$

$$\mathbf{P}(z) = \frac{0.1651z^{-1} - 0.1565z^{-2}}{1 - 1.9193z^{-1} + 0.9206z^{-2}}$$

采用改进型离散时间重复控制器，设计稳定化补偿器 $\mathbf{C}(z)$ ，有

$$\mathbf{C}(z) = 9.6712k_r (-0.1565z^2 + 0.4655z - 0.4610 + 0.1520z^{-1}), \quad k_r \in (0, 2)$$

对于薄板上点(0.0898, 0.0149)，低通滤波器 $\mathbf{F}(z) = 1/4(2 + z + z^{-1})$ 。控制对象离散状态方程中各矩阵和 z 域传递函数为

$$\mathbf{F}_d = \begin{bmatrix} 0.9980 & 0 & 0 & 0 \\ 0 & 0.9826 & 0 & 0 \\ -14.0500 & 0 & 0.9786 & 0 \\ 0 & -124.3340 & 0 & 0.9407 \end{bmatrix}$$

$$\mathbf{G}_d = \begin{bmatrix} 0 \\ 0 \\ 0.0186 \\ 0.0103 \end{bmatrix} \quad \mathbf{C}_d = [0 \quad 0 \quad 3.2043 \quad 2.2194]$$

$$\mathbf{P}(z) = \frac{0.0825z^{-1} - 0.0785z^{-2}}{1 - 1.9193z^{-1} + 0.9206z^{-2}}$$

采用改进型离散时间重复控制器，设计稳定化补偿器 $\mathbf{C}(z)$ ，有

$$\mathbf{C}(z) = 38.6100k_r (-0.0785z^2 + 0.2332z - 0.2306 + 0.0759z^{-1}), \quad k_r \in (0, 2)$$

(1) 当以 36 Hz 正弦，即以第一阶模态频率激励薄板时，激励信号如图6.22所示，控制系统速度响应和自功率谱如图6.23所示。

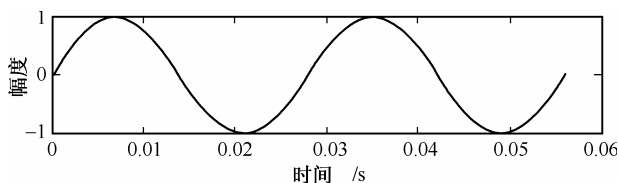


图 6.22 正弦激励信号

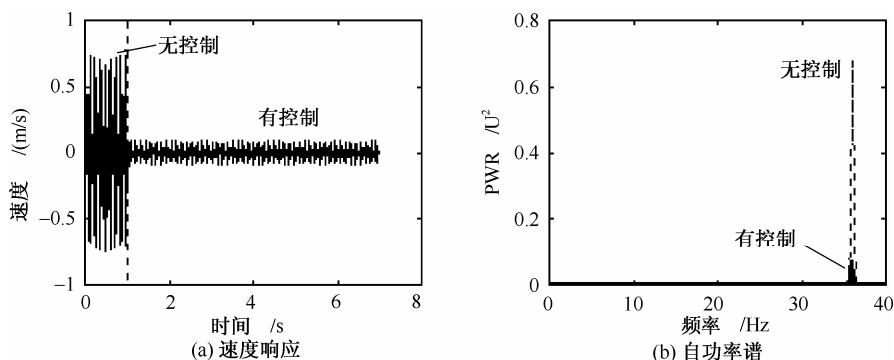


图 6.23 正弦激励时悬臂约束阻尼薄板离散时间重复振动控制系统效果

(2) 当以 36 Hz 和 108 Hz 复合正弦，即以第一阶和第二阶模态频率激励薄板时，激励信号如图6.24所示，控制系统速度响应和自功率谱如图6.25所示。

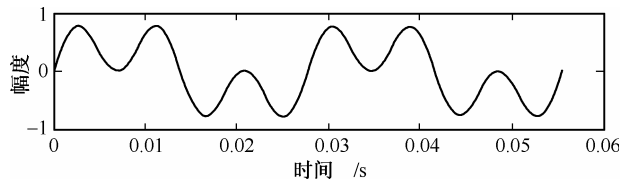


图 6.24 复合正弦激励信号

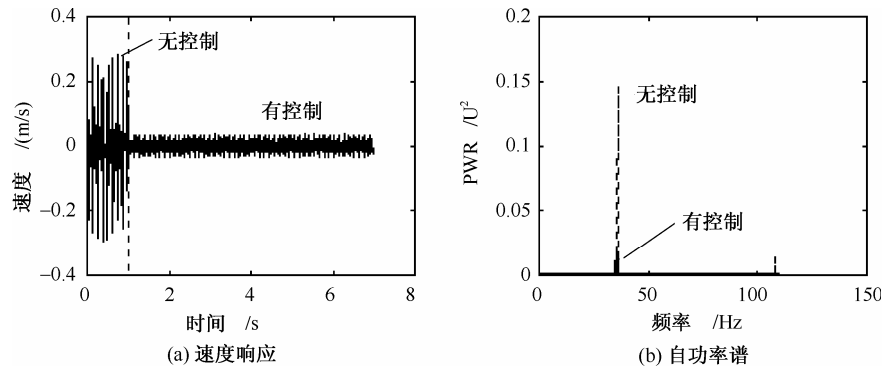


图 6.25 复合正弦激励时悬臂约束阻尼薄板离散时间重复振动控制系统效果

(3) 当以扫描时间 333 ms，扫描频率 36~46 Hz 的正弦扫描信号激励薄板时，激励信号如图6.26所示，控制系统速度响应和自功率谱如图6.27所示。

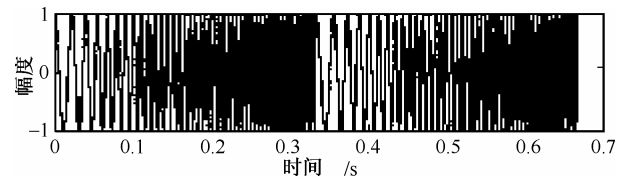


图 6.26 复合正弦激励信号

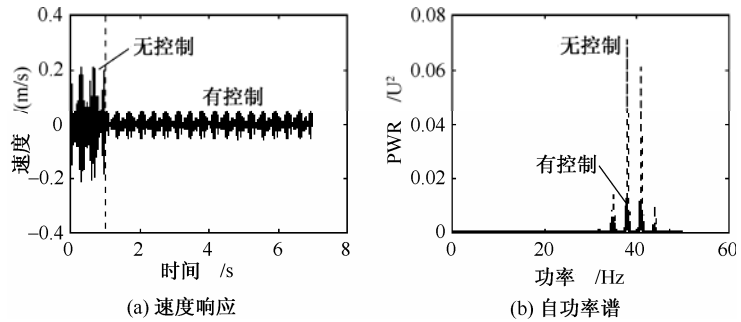


图 6.27 正弦扫描激励时悬臂约束阻尼薄板离散时间重复振动控制系统效果

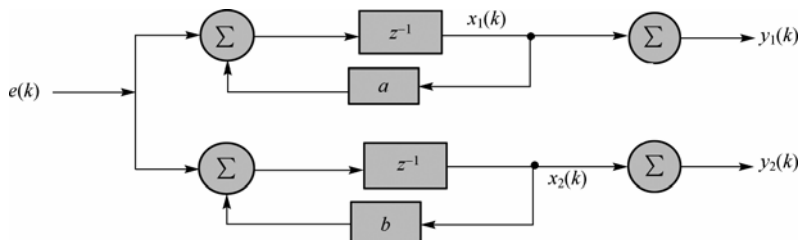
习题与思考题

- 1 什么叫状态变量？对实际系统来说，它的状态变量的选取是否唯一？
- 2 描述系统的差分方程为

$$y(n) + 2y(n-1) - 3y(n-2) + 4y(n-3) = x(n-1) + 2x(n-2) - 3x(n-3)$$

写出其状态方程和输出方程。

- 3 列写如图所示离散系统的状态方程和输出方程。



- 4 已知线性离散系统的离散状态方程

$$\mathbf{x}(kT+T) = \begin{bmatrix} 0 & -1 \\ -0.4 & 0.3 \end{bmatrix} \mathbf{x}(kT) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \mathbf{u}(kT), \quad \mathbf{y}(kT) = \begin{bmatrix} 0 & 1 \end{bmatrix} \mathbf{x}(kT)$$

$$\mathbf{x}(0) = \begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \quad \mathbf{u}(kT) = 1, \quad k \geq 0$$

(1) 试用迭代法, 求解 $\mathbf{x}(kT)$ 、 $\mathbf{y}(kT)$ 。

(2) 试用 Z 变换法, 求解 $\mathbf{x}(kT)$ 、 $\mathbf{y}(kT)$ 。

- 5 设线性离散系统的 Z 传递函数为

$$G(z) = \frac{2z^2 + 5z + 1}{z^2 + 3z + 2}$$

试求系统的离散状态空间表达式。

- 6 设离散时间系统的状态方程为

$$\mathbf{x}(k+1) = \begin{bmatrix} 0 & 1 \\ 0.16 & -1 \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \mathbf{u}(k)$$

$$\text{初始状态 } \mathbf{x}(0) = \begin{bmatrix} 1 \\ -1 \end{bmatrix},$$

试求系统在单位阶跃输入序列 (即 $\mathbf{u}(k) = 1, k = 0, 1, 2, \dots$) 作用下的状态响应 $\mathbf{x}(k), k = 0, 1, 2, \dots$ 。

- 7 已知离散时间系统的状态方程描述为

$$\mathbf{x}(k+1) = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 0 \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \mathbf{u}(k)$$

$$\mathbf{y}(k) = \begin{bmatrix} 1 & 1 \end{bmatrix} \mathbf{x}(k)$$

试求它的脉冲传递函数描述。

- 8 离散时间系统的能控性和能观测性是如何定义的? 它们之间有什么不同?

- 9 检验下列系统的能控性和能观测性

$$(1) \quad \mathbf{x}(k+1) = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \mathbf{u}(k) \quad \mathbf{y}(k) = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x}(k)$$

$$(2) \quad \mathbf{x}(k+1) = \begin{bmatrix} 0.1 & 1 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.5 \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \mathbf{u}(k) \quad \mathbf{y}(k) = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \mathbf{x}(k)$$

- 10 已知系统 $\begin{cases} \dot{\mathbf{x}}(k+1) = \begin{bmatrix} a & 1 \\ 0 & b \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \mathbf{u}(k) \\ \mathbf{y}(k) = \begin{bmatrix} 1 & -1 \end{bmatrix} \mathbf{x}(k) \end{cases}$ 的状态完全能控又能观测, 求参数 a, b 值。

第7章 智能控制技术

常用控制和近代控制理论在应用中都遇到不少难题，原因在于这些控制设计和分析的基础是数学模型，而实际存在的非线性、不确定性、时变性和不完全性等因素，使精确的系统数学模型难以获得。当研究这些控制时，必须提出的一些比较苛刻的假设往往与实际应用不相吻合。为了提高控制性能，整个系统变得较为复杂，可靠性降低，也增加了初始投资。

受到自然界发展规律的启迪，智能控制技术向生物学习，而不是根据复杂精确的数学模型来进行控制。智能控制通过模仿人的思维，利用人的知识和经验，模仿生物结构，模仿一些生物和自然现象的运行机制等方式，为解决上述问题找到了一条较好的出路。

智能控制交叉应用了计算机、生物学、数学、物理学、人工智能、自动控制、运筹学和信息论的最新研究成果，为解决复杂问题提供了新的思路 and 手段。

7.1 模糊控制

7.1.1 模糊数学基础

1. 二值逻辑与经典集合论

(1) 逻辑

庄子、亚里斯多德等古代先贤早就开始了对人类思维及其规律的探索。研究思维形式和规律的学科称为逻辑学。客观事物某些最普遍的性质在思维中的反映可归纳为如下形式逻辑的基本规律：

同一律 A 是 A

矛盾律 $A \vee \bar{A} \Leftrightarrow 1$

排中律 $A \wedge \bar{A} \Leftrightarrow 0$

矛盾律不允许同时既为真又为假，两个对立面不能重叠。排中律只允许是真或者是假，对立面应该是邻接毗连的。这两个定理合起来就是，整个论域只有两个对立面，对立面之间的界线是一根无宽度的几何线；黑白分明，非此即彼。

二值逻辑是以真、假为元素的数理逻辑。二值逻辑可以简单地用开关的“开”和“关”或者电平“高”与“低”来表示，其基本运算逻辑“补”、逻辑“与”和逻辑“或”也可以方便、容易地实现，因而在数字电子计算机上得到了成功的应用。加上它简单明了，在任何命题判断中都可以得到明确的结论，确实在整个科学发展过程中显示了强大的生命力。

(2) 经典集合

经典集合的定义是，有某种特定属性的对象的全体。

经典集合中，元素要么属于这个集合，要么就不属于这个集合，二者必居其一，这样可以用来明确地区分不同类的事物。其表示方法如下。

① 列举法。列举集合中所有元素，适用于元素有限的集合。

② 定义法。用集合中元素的共性来描述集合，适用于有很多元素但不能一一列举的集合。例如，所有奇数的集合表示为 $A=\{x|x \text{ 为奇数}\}$ 。

③ 特征函数法。利用经典集合非此即彼的明晰性来表示集合。例如，小于 10 的偶数构成的集合可表示成：若小于 10 的偶数 x 属于集合 A ，集合 A 就可以通过特征函数 $X_A(x)$ 来表示，即

$$X_A(x) = \begin{cases} 1, & x \in A \\ 0, & x \notin A \end{cases}$$

④ 扎德 (Zadeh) 表示法。例如，“小于 10 的偶数” $= 0/1 + 1/2 + 0/3 + 1/4 + 0/5 + 1/6 + 0/7 + 1/8 + 0/9$ ，式中借用了加号和分式的形式：分母表示集合中的元素，分子表示该元素的特征函数值，“+”表示列举。

(3) 推理与蕴含

① 推理是由一个或几个已知判断（前提）推出未知判断（结论）的思维形式。推理是客观事物的联系在人们意识中的反映。要使推理成立，即推理得到的间接知识是真实的，必须遵守两个条件：前提真实和推理的形式正确。

② 蕴含是数理逻辑中一种命题连接词，用符号“ \rightarrow ”表示。其定义为

$$A \rightarrow B \Leftrightarrow \bar{A} \vee B$$

$A \rightarrow B$ 即 A 和 B 的蕴含式，读做“ A 蕴含 B ”，其真假关系为若 A 真 B 假，则 $A \rightarrow B$ 假，否则 $A \rightarrow B$ 真； A 、 B 分别称为 $A \rightarrow B$ 的前件和后件。

由图 7.1 可以看出，如果 A 完全在 B 内，则 $\bar{A} \vee B = 1$ ；如果 A 不完全在 B 内，则 $\bar{A} \vee B \neq 1$ 。

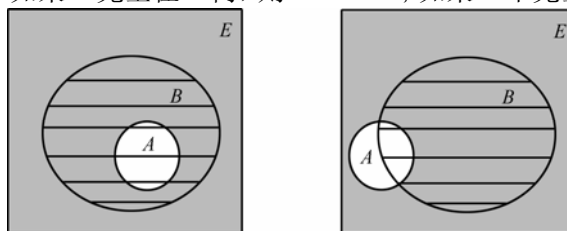


图 7.1 蕴含定义的图示

③ 推理规则。在以下推理规则中，用 $A_1, A_2, \dots, A_k | B$ 表示 B 是 A_1, A_2, \dots, A_k 的逻辑结论，即如果 A_1, A_2, \dots, A_k 成立，则形式上 B 也成立。

- 假言推理规则： $A \rightarrow B, A | B$
- 附加规则： $A | A \vee B$
- 化简规则： $A \wedge B | A$
- 拒取式规则： $A \rightarrow B, \bar{B} | \bar{A}$
- 假言三段论规则： $A \rightarrow B, B \rightarrow C | A \rightarrow C$
- 析取三段论规则： $A \vee B, \bar{B} | A$
- 构造性二难规则： $A \rightarrow B, C \rightarrow D, A \vee C | B \vee D$
- 合取引入规则： $A, B | A \wedge B$

著名的苏格拉底三段论就是利用假言三段论规则进行推理的（如图 7.2 所示）：

人都是要死的；苏格拉底是人；所以苏格拉底是要死的。
经典集合论充分体现了亚里斯多德二值逻辑，在数学上也不断获得了成功。

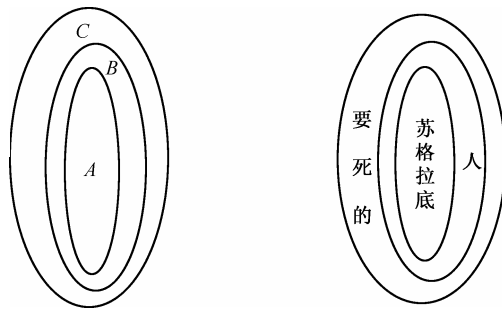


图 7.2 苏格拉底三段论

2. 从经典集合到模糊集合

(1) 经典集合的困惑

然而，进一步对周围的事物细加考察，就会发现经典集合并不能描述所有的事物，特别是涉及到与人的认识有关的概念和现象。

例如，从一堆沙子中一粒一粒地不断取沙子，什么时候留下的才能算是一堆呢？“舒服的温度”与相邻的“凉”和“热”、“红色”与相邻的“橙色”、“胖子”与相邻的“匀称”又怎样划分呢？经典集合陷入了困惑。

这是因为“一堆”、“舒服的温度”、“红色”、“胖子”等根据常识的这些描述，实际上都是一个没有明确界限的范围，而且与其相邻的描述状态事实上又是交叉的，因而界限是模糊的，当然都无法用经典集合来描述和划分。

(2) 人类思维与概念的模糊性

日常生活中，每天都会遇到大量的模糊概念和模糊现象。这些模糊概念都无法用传统的数学方法来度量，但是大家都能心领神会；相反，如果给出精确的描述，得到的结果倒可能是模糊的。例如，“太少”、“较大”、“小了一些”、“很高”等都是模糊概念。我们很容易在人群中找出一个“秃头”，但几乎没有人知道谁的头发是多少根。

根据模糊概念进行控制，却可以得到精确的结果，这种人的思维方式与传统数学精确表达方式的不一致，大家也已习以为常。例如，很难根据精确的方程来学习骑自行车，但如果遵循“自行车往哪边倒，龙头（车把）就往哪边拐；倒得越多，拐得越大”这样简单的模糊规则，很快就能保持自行车的平衡。

这说明，模糊性是人们思维方式的一种反映，是人类思维和表达的重要特征之一。模糊的定性描述利用了人们原有的知识，大大地增加了信息量；也就是说，模糊概念比精确描述可以覆盖更广的范围。

(3) 把经典集合扩充成模糊集合

为了描述和处理人的自然语言或概念，扎德提出了一种新的数学工具：模糊集合（Fuzzy Sets）。经典集合对事物只用“1”和“0”简单地表示“属于”和“不属于”的分类；模糊集合则用从 0 到 1 之间的连续变化值来描述命题的“真（白）”和“假（黑）”之间逐渐变化的无穷多层次的灰度，这种连续变化值被称为隶属度（Degree of Membership）。

隶属度描述了元素属于该集合的程度。也就是说，有些东西可以同时是部分真和部分假；

一个元素也可以部分属于某个集合，部分属于另一个集合。其实日常语言中也常有这样的描述，例如，说某某“三分像人，七分像鬼”。

3. 模糊集合和隶属函数

(1) 定义

论域 U 中的模糊集合 F 是指，对于论域 U 的任一元素 $u \in U$ ，都指定了闭区间 $[0, 1]$ 中的一个数 $\mu_F(u) \in [0, 1]$ 与之对应， $\mu_F(u)$ 称为 u 对 A 的隶属度。

$\mu_F(u)$ 说明了 u 隶属于 U 的程度： $\mu_F(u) = 1$ ，表示 u 完全属于 U ； $\mu_F(u) = 0$ ，表示 u 完全不属于 U ； $0 < \mu_F(u) < 1$ ，表示 u 部分属于 U 。

由此定义的映射 μ_F 称为模糊集合 F 的隶属函数 (Membership Function)。

模糊集合完全由其隶属函数所刻画，模糊命题通过隶属函数建立了语言变量的数学表示。例如，将年轻、中年、老年这三个年龄的特征分别用模糊集合 A, B, C 表示，它们的论域都是 $U = [1, 100]$ ，论域中的元素是年龄 u ，规定模糊集合 A, B, C 的隶属函数 $\mu_A(u), \mu_B(u), \mu_C(u)$ 如图7.3所示。

设 $u_1 = 30$ ， u_1 对 A 的隶属度 $\mu_A(u_1) = 0.75$ ，这意味着 30 岁属于“年轻”的程度是 0.75。
设 $u_2 = 40$ ， u_2 既属于 A 又属于 B ， $\mu_A(u_2) = 0.25$ ， $\mu_B(u_2) = 0.5$ ，这说明 40 岁的人已不太年轻了，比较接近中年，但属于“中年”的程度还不太大，只有 0.5。
设 $u_3 = 52$ ， $\mu_B(u_3) = 1$ ， $\mu_C(u_3) = 0.1$ ，这说明 52 岁正值“中年”，但也已开始走向“老年”。
显然，模糊集合比较准确地、真实地描述了人们头脑中的原有概念；而将年龄段用阈值绝对地划分，用普通集合来描述模糊性概念是粗糙的。

模糊集合的隶属函数是经典集合特征函数的扩展和一般化。如图7.4所示，经典集合中，特征函数与 $\{0, 1\}$ 对应，而模糊集合中，特征函数的取值范围从两个元素的集合扩大到在 $[0, 1]$ 区间连续取值。

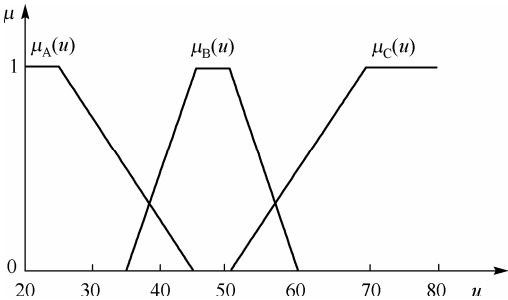


图 7.3 年轻、中年、老年的隶属函数

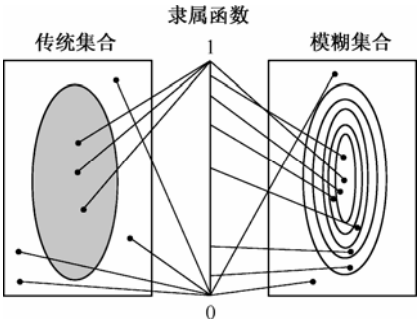


图 7.4 经典集合和模糊集合的映射

(2) 模糊集合的表示方法

① 连续论域。如果论域 U 是实数域，论域中有无穷多个连续的点，该论域称为连续论域。连续论域上的模糊集合可表示为

$$F = \int_{u \in U} \mu_F(u) / u \tag{7.1}$$

式中， \int 并不表示通常的积分，只是表示对论域中的每个元素 u 都定义了相应的隶属函数 $\mu_F(u)$ 。

常用的隶属函数曲线除折线型（如图7.3所示）外，还有单点型、三角型、正态型、矩型、

双峰型等，如图7.5所示。

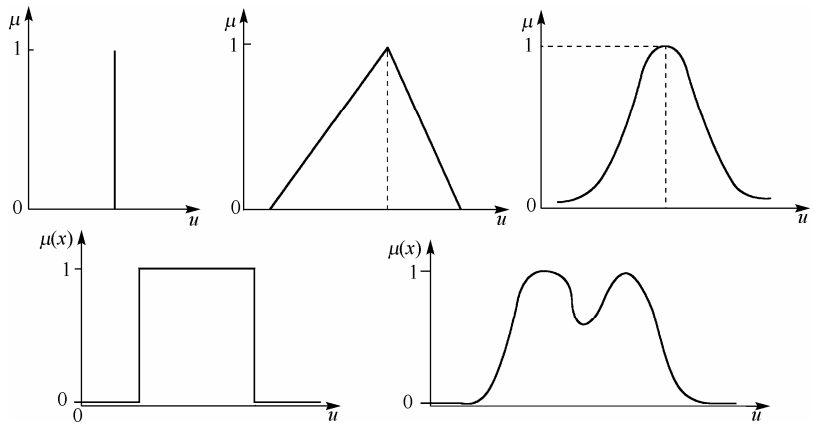


图 7.5 隶属函数曲线

② 离散论域。如果论域 U 中只包含有限个元素，则该论域称为离散论域。设离散论域 $U = \{u_1, u_2, \dots, u_n\}$ ， U 上的模糊集合 F 可表示为

$$F = \sum_{i=1}^n \mu_F(u_i) / u_i = \mu_F(u_1) / u_1 + \mu_F(u_2) / u_2 + \dots + \mu_F(u_n) / u_n \tag{7.2}$$

类似于经典集合的扎德表示法，式中借用了 Σ 、加号和分式形式： Σ 并不表示“求和”，而只是表示集合的一种方法；“+”表示列举；分母表示集合中的元素，分子表示该元素的特征函数值。例如，把常温范围分成含五个元素的集合：常温范围 = $\{0^\circ\text{C}, 10^\circ\text{C}, 20^\circ\text{C}, 30^\circ\text{C}, 40^\circ\text{C}\}$ 。

设“舒适温度”的隶属函数为 $\mu_M(x)$ ，则用 M 表示“舒适温度”的模糊命题一般形式为

$$M = \sum \mu_M(x_i) / x_i = 0.25/0 + 0.5/10 + 1.0/20 + 0.5/30 + 0.25/40$$

这里分式的含义是，分母表示温度值，分子表示该温度值对“舒适温度”的隶属度。

(3) 隶属度和概率

曾经有人认为模糊逻辑是概率理论的改头换面，也有人对隶属度和概率的概念混淆不清，但两者有着本质的不同。隶属度是描述模糊性的基本工具，概率是描述随机性的基本工具。

模糊性主要是人为的主观理解上的不确定性，通常是指对概念的定义及语言意义理解上的不确定性。例如，“老人”、“温度高”、“数量大”等所含的不确定性即为模糊性。

随机性则主要反映的是客观上自然的不确定性，或者是事件发生的偶然性。

模糊性和随机性是不同情况下的不确定性。例如，“明天有雨”的不确定性，是由今天的预测产生的，但到明天就变成确定的了；“掷一下骰子出几点”的不确定性是由于掷骰子之前推测产生的，掷下骰子后，几点就是确定的事件了。而“老人”、“气温高”等的不确定性来自语言意义上的模糊性，即使过一段时间，做了实验，它仍然是不确定的。

隶属度和概率两者不能相互替代，但可能相互结合，用来描述模糊随机事件的概率，即模糊概率。例如，求“明天下大雨”的概率，其中“下大雨”便是模糊事件。

(4) 确定隶属函数的方法

目前，确定隶属函数的方法还不成熟，隶属函数的确定也不是唯一的。

① 经验法。依靠经验确定隶属函数，然后再根据实验、试验或者计算机模拟得到的反馈

信息进行修正，这种没有理论化的方法带有一定的盲目性和主观性。即使是根据专家经验确定的隶属函数，也不能从理论上保证其正确性。

② 模糊统计法。通过对足够多人的调查，在讨论的论域中对要确定的模糊概念逐一写出定量范围，再进行统计处理，以确定能被大多数人认可的隶属函数。在一定条件下，随着调查人数的增加，各个元素隶属度都趋于一个稳定的值。然而，这种方法工作量大，一般很难在实际中采用。

③ 神经网络自动生成法。通过对神经网络的训练，利用神经网络的自学习功能，直接自动地生成隶属函数和规则。

4. 模糊集合的基本运算

(1) 模糊集合基本运算定义

设 A, B 为同一论域 U 中两个模糊集合，隶属函数分别为 μ_A 和 μ_B ，则模糊集合理论中的相等、包含和交、并、补等运算可通过它们的隶属函数来定义。

● A 与 B 相等，记为 $A=B$ ，有

$$\mu_A(u) = \mu_B(u) \quad \forall u \in U \tag{7.3}$$

● A 包含 B ，记为 $A \supseteq B$ ，有

$$\mu_A(u) \geq \mu_B(u) \quad \forall u \in U \tag{7.4}$$

● A 与 B 的交，记为 $A \cap B$ ，有

$$\mu_{A \cap B}(u) = \mu_A(u) \wedge \mu_B(u) = \min \{ \mu_A(u), \mu_B(u) \}, \quad \forall u \in U \tag{7.5}$$

● A 与 B 的并，记为 $A \cup B$ ，有

$$\mu_{A \cup B}(u) = \mu_A(u) \vee \mu_B(u) = \max \{ \mu_A(u), \mu_B(u) \}, \quad \forall u \in U \tag{7.6}$$

● A 的补，记为 \bar{A} ，有

$$\mu_{\bar{A}}(u) = 1 - \mu_A(u), \quad \forall u \in U \tag{7.7}$$

其中， \max 和 \vee 表示取大运算， \min 和 \wedge 表示取小运算。交、并、补三种运算对应的隶属函数图形分别如图7.6(a)、(b)、(c)所示。

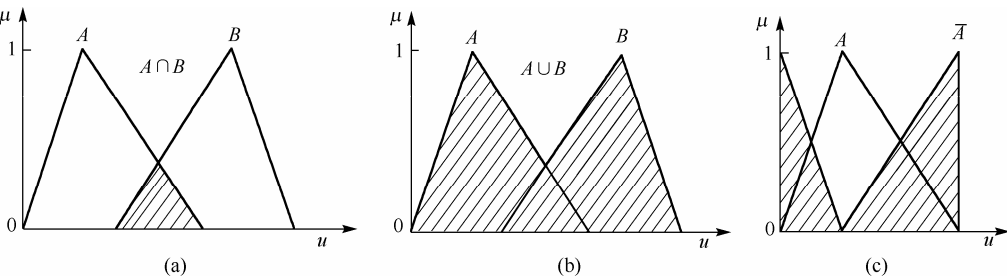


图 7.6 模糊集合的三种运算

模糊集合之间的并和交运算可以推广到 U 上的 n 个模糊集合。

(2) 模糊集合基本运算定律

和普通集合完全相同，模糊集合也有恒等律、交换律、结合律、分配律、吸收律、同一律、复原律、对偶律等八条运算定律。但由图7.7可以清楚地看出，模糊集合的运算不满足“互补律”，即

$$A \cup \bar{A} \neq 1; A \cap \bar{A} \neq 0 \quad (7.8)$$

(3) α 截集与分解定理

α 截集将隶属函数按下式转化为特征函数 (如图7.8 所示):

$$C_{A\lambda}(x) = \begin{cases} 1, & \text{当 } \mu_A(x) \geq \lambda \text{ 时} \\ 0, & \text{当 } \mu_A(x) < \lambda \text{ 时} \end{cases} \quad (7.9)$$

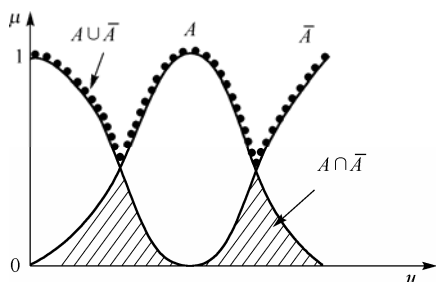


图 7.7 模糊集合的运算不满足互补律

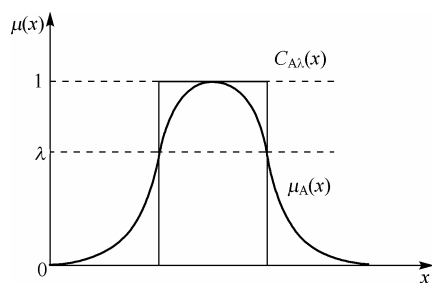


图 7.8 α 截集

分解定理如图7.9所示。

设 $A \in \Gamma(x)$, A_λ 是 A 的 λ 截集, $\lambda \in [0, 1]$

则

$$A = \bigcup_{\lambda \in [0, 1]} \lambda A_\lambda$$

其中

$$\begin{aligned} \lambda A_\lambda &\in \Gamma(x) \\ \mu_{\lambda A_\lambda}(x) &= \begin{cases} \lambda, & \text{当 } x \in A_\lambda \\ 0, & \text{当 } x \notin A_\lambda \end{cases} \\ \mu_{\bigcup_{\lambda \in [0, 1]} \lambda A_\lambda}(x) &= \sup_{\lambda \in [0, 1]} \mu_{\lambda A_\lambda}(x) \end{aligned} \quad (7.10)$$

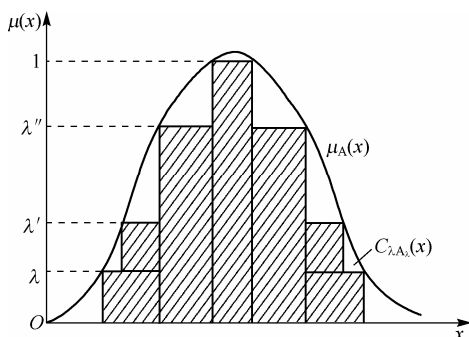


图 7.9 分解定理

7.1.2 模糊关系与模糊推理

1. 模糊关系

(1) 关系

关系是描述客观世界的事物之间联系的数学模型之一, 常用符号 R 表示。

若 R 为由集合 X 到集合 Y 的普通关系, 则对任意 $x \in X$, $y \in Y$ 都只能有以下两种情况: x 与 y 有某种关系, 即 xRy ; 或者, x 与 y 无某种关系, 即 $x\bar{R}y$ 。

由 X 到 Y 的某种关系 R , 也可用序对 (x, y) 来表示, 其中 $x \in X$, $y \in Y$ 。所有有关系 R 的序对可以构成一个 R 集。

(2) 模糊关系的定义

两组事物之间的关系不宜用有或无来进行肯定或否定的回答时, 可以用模糊关系来描述。

设 $X \times Y$ 是集合 X 与集合 Y 的直积, 以 $X \times Y$ 为论域定义的模糊集合 R 称为 X 和 Y 的模糊关系。也就是说, 对 $X \times Y$ 中的任一元素 (x, y) , 都指定了它对 R 的隶属度 $\mu_R(x, y)$, R 的隶属函数 μ_R 可看做是如下的映射:

$$\begin{aligned}\mu_R : X \times Y &\rightarrow [0, 1] \\ (x, y) &\mapsto \mu_R(x, y)\end{aligned}$$

模糊关系同样有自返性、对称性、传递性和等价性等关系。

模糊关系的特性也完全由隶属函数来刻画， $\mu_R(x, y)$ 是以 x, y 为自变量的一个空间曲面。

当 X 和 Y 皆为有限的离散集合时， X 和 Y 的模糊关系 R 可用矩阵表示，称为模糊关系矩阵，即

$$R_{X \times Y} = (a_{ij})_{m \times n} = (\mu_R(x_i, y_j))_{m \times n}, \quad i = 1, 2, \dots, m; \quad j = 1, 2, \dots, n \quad (7.11)$$

当用模糊矩阵来表示模糊关系时，矩阵中的 a_{ij} 表示集合 X 中的第 i 个元素 x_i 和集合 Y 中的第 j 个元素 y_j 隶属于模糊关系 R 的程度，记为 $\mu_R(x, y)$ ， $\mu_R(x, y)$ 在闭区间 $[0, 1]$ 中取值。

模糊矩阵的一般形式为

$$R = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \quad (7.12)$$

其中， $0 \leq a_{ij} \leq 1$ ， $1 \leq i \leq m$ ， $1 \leq j \leq n$ 。

例如，设 X 为横轴， Y 为纵轴，直积 $X \times Y$ 即整个平面。模糊关系“ x 远远大于 y ”的隶属函数确定为

$$\mu_R(x, y) = \begin{cases} 0 & \forall x \leq y \\ \frac{1}{1 + \frac{100}{(x - y)^2}} & \forall x > y \end{cases}$$

在 X 中取 10, 20, 40, 80 四个点， Y 中取 17, 20, 30, 40 四个点，则“ x 远远大于 y ”的模糊关系矩阵为

$$R = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0.08 & 0 & 0 & 0 \\ 0.84 & 0.8 & 0.5 & 0 \\ 0.98 & 0.97 & 0.96 & 0.94 \end{bmatrix}$$

当 $x=40$, $y=20$ 时，“ x 远远大于 y ”的程度是 0.8。

(3) 模糊关系的运算

除了与一般模糊集合完全相同的相等、包含和并、交、补等运算以外，模糊关系还有合成运算、幂运算、逆运算等特殊的运算。

① 并运算。对于 $A=[a_{ij}]$ 和 $B=[b_{ij}]$ ，若有 $c_{ij} = \max[a_{ij}, b_{ij}] = a_{ij} \vee b_{ij}$ ，则称 $C=[c_{ij}]$ 为 A 和 B 的并，记为 $C = A \cup B$ 。

② 交运算。对于 $A=[a_{ij}]$ 和 $B=[b_{ij}]$ ，若有 $c_{ij} = \min[a_{ij}, b_{ij}] = a_{ij} \wedge b_{ij}$ ，则称 $C=[c_{ij}]$ 为 A 和 B 的交，记为 $C = A \cap B$ 。

③ 补运算。对于 $A=[a_{ij}]$ ，则 $[1 - a_{ij}]$ 为 A 的补矩阵，记为 \bar{A} 。

④ 合成运算。设 A 是 X 到 Y 的模糊关系, B 是 Y 到 Z 的模糊关系, 则 A 和 B 的合成是 X 到 Z 的一个模糊关系, 记为 $A \circ B$, 扎德将其隶属函数定义为

$$\mu_{A \circ B}(x, z) = \vee [\mu_A(x, y) \wedge \mu_B(y, z)], \quad \forall (x, z) \in X \times Z \quad (7.13)$$

可以看出, 模糊矩阵的合成运算与普通矩阵的乘法运算类似, 所不同的是并非两项相乘后再相加, 而是先取小后取大。

若 $C = A \circ B$, 则 C 中的元素

$$c_{ij} = \max \{ \min [a_{ik}, b_{kj}] \} = \vee [a_{ik} \wedge b_{kj}] \quad (7.14)$$

例如:

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}, \quad B = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$$

则

$$A \square B = \begin{bmatrix} (a_{11} \wedge b_{11}) \vee (a_{12} \wedge b_{21}) & (a_{11} \wedge b_{12}) \vee (a_{12} \wedge b_{22}) \\ (a_{21} \wedge b_{11}) \vee (a_{22} \wedge b_{21}) & (a_{21} \wedge b_{12}) \vee (a_{22} \wedge b_{22}) \end{bmatrix}$$

⑤ 幂运算。设 R 是 $X \times X$ 上的模糊关系, 则它的模糊关系矩阵为方阵, R 的幂定义为

$$R^n = R \circ R \circ \dots \circ R \quad (n \text{ 个 } R \text{ 合成})$$

模糊关系矩阵的幂满足以下指数法则:

$$\begin{aligned} R^m \circ R^n &= R^{m+n} \\ (R^m)^n &= R^{mn} \end{aligned}$$

⑥ 逆运算。设 R 是 X 到 Y 的模糊关系, 则其逆模糊关系 R^{-1} 是 Y 到 X 的一个模糊关系, 其隶属函数为

$$\mu_{R^{-1}}(y, x) = \mu_R(x, y), \quad \forall (y, x) \in Y \times X \quad (7.15)$$

例如, 模糊关系“ x 远远大于 y ”的逆模糊关系为“ y 远远小于 x ”, 对于 Y 中取 17, 20, 30, 40 四个点, 在 X 中取 10, 20, 40, 80 四个点, “ y 远远小于 x ”的隶属度函数为

$$R^{-1} = \begin{bmatrix} 0 & 0.08 & 0.84 & 0.98 \\ 0 & 0 & 0.8 & 0.97 \\ 0 & 0 & 0.5 & 0.96 \\ 0 & 0 & 0 & 0.94 \end{bmatrix} = R^T$$

2. 模糊蕴含与模糊推理

人们获取的信息往往不精确, 不完全, 或者事实本身就是模糊而不完全确定, 但又必须这样的环境中进行判断和决策。

在传统形式逻辑的确定性推理基础上发展起来的不确定性推理是人工智能的重要课题之一。模糊推理就是一种不确定性推理方法。

(1) 模糊命题与模糊蕴含

模糊命题通过隶属函数建立了语言变量的数学表示。

① 普通模糊命题。含有模糊概念的简单陈述句称为普通模糊命题。例如，今天天气热。语言变量是气温，“热”是论域 $[-40\text{ }^{\circ}\text{C}, 50\text{ }^{\circ}\text{C}]$ 上的模糊集合，隶属函数为 $\mu_H(t)$ 。若今日气温为 $t_0=20\text{ }^{\circ}\text{C}$ ， $\mu_H(20)=0.4$ ，那么该命题“今天天气热”的真实程度是0.4。

② 条件模糊命题。条件模糊命题表达了两个普通模糊命题之间的因果关系：如果 x 为 A ，则 y 为 B 。例如，如果 x 小，则 y 大；如果温度高，那么压力就大。

上述条件模糊命题可简写为“如果 P 为真，则 Q 为真”，即 $P\rightarrow Q$ ，它表示普通模糊命题 P 、 Q 之间有因果关系， P 、 Q 的真实程度分别用隶属函数 $\mu_A(x)$ 和 $\mu_B(y)$ 表示。此时不同论域上的模糊集合 A 与 B 之间的模糊关系称为模糊蕴含 $R_{A\rightarrow B}$ 。

模糊蕴含 $R_{A\rightarrow B}$ 有多种定义，例如：

仿照经典集合中的蕴含运算，一种最大最小算术运算模糊蕴含定义为

$$R_{A\rightarrow B} = (A \wedge B) \vee (1 - A) \tag{7.16}$$

其隶属函数为

$$\mu_{A\rightarrow B}(x, y) = [\mu_A(x) \wedge \mu_B(y)] \vee [1 - \mu_A(x)] \tag{7.17}$$

模糊控制中常用的最小运算模糊蕴含定义为

$$R_{A\rightarrow B} = A \wedge B \tag{7.18}$$

其隶属函数为

$$\mu_{A\rightarrow B}(x, y) = \mu_A(x) \wedge \mu_B(y) \tag{7.19}$$

【例 7.1】 设论域 $X = \{a_1, a_2, a_3, a_4, a_5\}$ ， $Y = \{b_1, b_2, b_3, b_4\}$ 上的模糊集合分别为

$$\begin{aligned} \text{“小”} = A &= \frac{1}{a_1} + \frac{0.8}{a_2} + \frac{0.4}{a_3} + \frac{0}{a_4} + \frac{0}{a_5} \\ \text{“大”} = B &= \frac{0}{b_1} + \frac{0}{b_2} + \frac{0.3}{b_3} + \frac{0.9}{b_4} \end{aligned}$$

根据最大最小算术运算模糊蕴含规则，“如果 x 为小，则 y 为大”的模糊关系矩阵 $R_{A\rightarrow B}$ 等于：

$$\begin{aligned} R_{A\rightarrow B} &= \begin{bmatrix} 1 \\ 0.8 \\ 0.4 \\ 0 \\ 0 \end{bmatrix} \wedge \begin{bmatrix} 0 & 0 & 0.3 & 0.9 \end{bmatrix} \vee \begin{bmatrix} 0 \\ 0.2 \\ 0.6 \\ 1 \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 & 0.3 & 0.9 \\ 0 & 0 & 0.3 & 0.8 \\ 0 & 0 & 0.3 & 0.4 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \vee \begin{bmatrix} 0 \\ 0.2 \\ 0.6 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0.3 & 0.9 \\ 0.2 & 0.2 & 0.3 & 0.8 \\ 0.6 & 0.6 & 0.6 & 0.6 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \end{aligned}$$

(2) 语言算子

自然语言中有许多修饰词，如“较”、“很”、“大约”、“倾向于”等。为了对模糊的自然语言形式化和定量化，引入语言算子来进一步区分和刻划模糊概念程度。描述模糊值的语言算子可分为三类：语气算子、模糊化算子和判定化算子。

① 语气算子。用于表达模糊值的肯定程度，是对原隶属函数乘 α 次方，即 μ^α 。其中：

- 集中化算子， $\alpha > 1$ ：起加强语气作用，如“很”、“极”、“非常”；
- 散漫化算子， $\alpha < 1$ ：起削弱语气作用，如“稍”、“比较”、“略微”。

【例 7.2】若年老的隶属函数定义为

$$\mu_{\text{年老}}(x) = \begin{cases} 0 & 0 \leq x \leq 50 \\ \frac{1}{1 + (\frac{5}{x-50})^2} & x > 50 \end{cases}$$

则

$$\begin{aligned} \mu_{\text{极老}}(x) &= \mu_{\text{年老}}^4(x) & \mu_{\text{比较老}}(x) &= \mu_{\text{年老}}^{3/4}(x) \\ \mu_{\text{很老}}(x) &= \mu_{\text{年老}}^2(x) & \mu_{\text{略老}}(x) &= \mu_{\text{年老}}^{1/2}(x) \\ \mu_{\text{相当老}}(x) &= \mu_{\text{年老}}^{5/4}(x) & \mu_{\text{稍老}}(x) &= \mu_{\text{年老}}^{1/4}(x) \end{aligned}$$

由于隶属函数的取值范围在闭区间 $[0, 1]$ ，集中化算子将使隶属函数曲线趋于尖锐化，而且幂指数越大 >1 ，曲线越尖锐；反之，散漫化算子将使隶属函数曲线趋于平坦化，而且幂指数越小 <1 ，曲线越平坦。

② 模糊化算子。用于把肯定转化为模糊，例如，“大约”、“近似”。如果对数字进行作用，就把精确数转化为模糊数，例如，数字“65”是精确数，而“大约 65”就是模糊数。

③ 判定化算子。用于把模糊值进行肯定化处理，对模糊值做出倾向性判断，例如，“倾向于”、“偏向于”等。其对数字处理方法类似于“四舍五入”，并常把隶属度为 0.5 作为分界。

(3) 模糊推理

常用的模糊推理方法有两种：广义前向推理（Generalized Modus Ponens）和广义反向推理（Generalized Modus Tollens），它们都是通常所说的“三段论”。前提 1 就是一条“IF…，THEN…”形式的条件模糊命题，IF 部分是规则的前提，THEN 部分是规则的结论，其推理过程如下。

广义前向推理（GMP）的推理过程是由已知规则的前提求结论：

前提 1：如果 x 为 A ，则 y 为 B ；前提 2： x 为 A' ；结论： y 为 B' 。

例如，如果秋冬的雨雪多，则来年的冬小麦收成好；今年秋冬的雨雪较多，那么明年冬小麦的收成如何？这属于模糊预报问题。

广义反向推理（GMT）的推理过程是由已知规则的结论求前提：

前提 1：如果 x 为 A ，则 y 为 B ；前提 2： y 为 B' ；结论： x 为 A' 。

例如，如果患了肝炎，则 GPT 指标高；某人的 GPT 指标不很高，那么他患肝炎的可能性有多大？这属于模糊诊断问题。

这里前提 2 中的模糊判断和前提 1 中前件或后件并不严格相同而是相近，它们有程度上的差别，因而推理过程具有模糊性，具体表现在推理规则是模糊的。

● 广义前向推理: $B' = A' \circ R_{A \rightarrow B}$;

● 广义反向推理: $A' = R_{A \rightarrow B} \circ B'$ 。

【例 7.3】若 x 小则 y 大, 已给 x 较小, 问 y 如何?

设论域 $X = \{1, 2, 3, 4, 5\}$, $Y = \{1, 2, 3, 4\}$

$$[\text{小}] = A = \frac{1}{1} + \frac{0.8}{2} + \frac{0.4}{3} + \frac{0}{4} + \frac{0}{5}$$

$$[\text{较小}] = A' = \frac{1}{1} + \frac{0.6}{2} + \frac{0.2}{3} + \frac{0.1}{4} + \frac{0}{5}$$

$$[\text{大}] = B = \frac{0}{1} + \frac{0}{2} + \frac{0.3}{3} + \frac{0.9}{4}$$

根据最大最小算术运算模糊蕴含定义, “如果 x 为小, 则 y 为大” 的模糊关系矩阵 (见例 7.1) 为

$$R_{A \rightarrow B} = \begin{bmatrix} 0 & 0 & 0.3 & 0.9 \\ 0.2 & 0.2 & 0.3 & 0.8 \\ 0.6 & 0.6 & 0.6 & 0.6 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

$$B' = A' \circ R_{A \rightarrow B} = [1 \quad 0.6 \quad 0.2 \quad 0.1 \quad 0] \circ \begin{bmatrix} 0 & 0 & 0.3 & 0.9 \\ 0.2 & 0.2 & 0.3 & 0.8 \\ 0.6 & 0.6 & 0.6 & 0.6 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

$$= [0.2 \quad 0.2 \quad 0.3 \quad 0.9]$$

这与我们推想到的“若 x 小则 y 大, 现 x 较小, 则 y 较大” 的结果是一致的。

根据最小运算模糊蕴含定义, 推理结论 B' 的隶属函数为

$$\mu_{B'}(y) = \bigcup_x \{ \mu_{A'}(x) \wedge [\mu_A(x) \wedge \mu_B(y)] \} = \bigcup_x \{ \mu_{A'}(x) \wedge \mu_A(x) \} \wedge \mu_B(y) = a \wedge \mu_B(y)$$

式中, $a = \bigcup_x \{ \mu_{A'}(x) \wedge \mu_A(x) \} = \text{Height}(A' \cap A)$ 是指模糊集合交集的高度, 也可以看成是 A' 对 A 的适配程度, 即隶属度。

根据最小运算方法, 结论 B' 由 a 与模糊集合 B 作模糊“与”即取小运算得到, 在图形上就是用 a 作基准对 B 削顶 (如图 7.10 所示)。

【例 7.4】若温度高, 则压力大, 如果温度较高, 问压力如何?

设论域 T (温度) = $\{0, 20, 40, 60, 80, 100\}$, P (压力) = $\{1, 2, 3, 4, 5, 6, 7\}$

$$\mu_A(\text{温度高}) = \frac{0}{0} + \frac{0.1}{20} + \frac{0.3}{40} + \frac{0.6}{60} + \frac{0.85}{80} + \frac{1}{100}$$

$$\mu_B(\text{压力大}) = \frac{0}{1} + \frac{0.1}{2} + \frac{0.3}{3} + \frac{0.5}{4} + \frac{0.7}{5} + \frac{0.85}{6} + \frac{1}{7}$$

$$\mu_{A'}(\text{温度较高}) = \frac{0.1}{0} + \frac{0.15}{20} + \frac{0.4}{40} + \frac{0.75}{60} + \frac{1}{80} + \frac{0.8}{100}$$

$$\begin{aligned} a &= \text{Height}(A' \cap A) \\ &= \text{Height}\left(\frac{0.1 \wedge 0}{0} + \frac{0.15 \wedge 0.1}{20} + \frac{0.4 \wedge 0.3}{40} + \frac{0.75 \wedge 0.6}{60} + \frac{1 \wedge 0.85}{80} + \frac{0.8 \wedge 1}{100}\right) = 0.85 \\ \mu_{B'} &= a \wedge \mu_B = 0.85 \wedge \left(\frac{0}{1} + \frac{0.1}{2} + \frac{0.3}{3} + \frac{0.5}{4} + \frac{0.7}{5} + \frac{0.85}{6} + \frac{1}{7}\right) \\ &= \left(\frac{0}{1} + \frac{0.1}{2} + \frac{0.3}{3} + \frac{0.5}{4} + \frac{0.7}{5} + \frac{0.85}{6} + \frac{0.85}{7}\right) \end{aligned}$$

可以认为近似于“压力较大”的隶属函数。

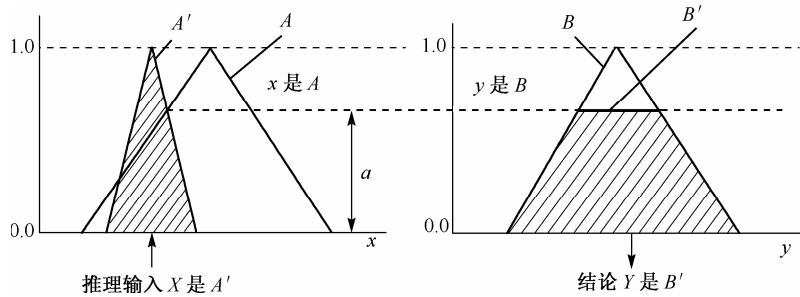


图 7.10 最小运算推理

7.1.3 模糊控制原理

模糊控制系统与通常计算机控制系统一样，都可以使用以误差作为输入的闭环反馈控制，其不同点在于：

- 经典控制的控制量为精确量，而模糊控制为模糊量；
- 经典控制要建立数学模型，而模糊控制以模糊规则进行模糊推理。

模糊系统的性能主要取决于结构、所采用的模糊规则、合成推理算法；以及模糊决策的方法等因素。模糊控制器则是模糊控制系统的核心，如图7.11所示：

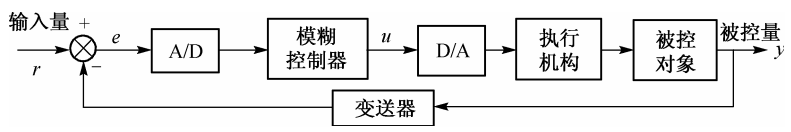


图 7.11 模糊控制系统的组成

1. 模糊控制器基本构成

模糊控制器主要包括输入模糊化接口、知识库、模糊推理、解模糊输出接口（非模糊化处理）四部分，如图7.12所示。

(1) 模糊化接口

测量输入变量（设定输入）和受控系统的输出变量，并把它们映射到一个合适的响应论域的量程，将精确的输入数据变换为适当的语言值或模糊集合的标志符。

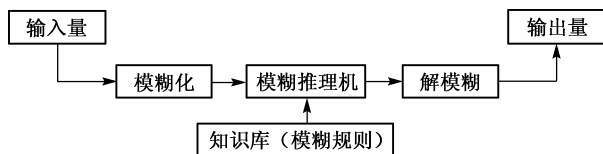


图 7.12 模糊控制器基本结构

(2) 知识库

由数据库和语言（模糊）控制规则库组成：数据库定义语言控制规则和模糊数据操作，语言控制规则表征控制目标和应用领域专家的控制策略。

(3) 推理机

是模糊控制系统的核心，以模糊概念为基础，模糊控制信息可通过模糊蕴含和模糊逻辑的推理规则来获取，并可实现拟人决策过程，根据模糊输入和模糊控制规则，按模糊推理合成规则计算模糊控制量，获得模糊输出。

(4) 解模糊输出接口

将得到的模糊控制量精确化后输出，通过量程变换进行输出定标，产生一个精确的控制作用。

2. 模糊控制器结构的选择

设计模糊控制器的第一步是选择模糊控制器的结构，也就是确定模糊控制器的输入变量和输出变量。

(1) 单输入单输出结构

假设模糊控制器输入变量为 X ，输出变量为 Y ，此时的模糊规则（ X 一般为控制误差， Y 为控制量）为

$$\begin{aligned}
 R_1: & \text{ if } X \text{ is } A_1 \text{ then } Y \text{ is } B_1 \\
 & \vdots \\
 R_n: & \text{ if } X \text{ is } A_n \text{ then } Y \text{ is } B_n
 \end{aligned}$$

其中， A_1, \dots, A_n 和 B_1, \dots, B_n 分别为输入和输出论域上的模糊子集。这类模糊规则的模糊关系为

$$R_{(x,y)} = \bigcup_{i=1}^n A_i \times B_i$$

(2) 双输入单输出结构

双输入单输出模糊控制的一般形式为

$$R_i: \text{ if } X_1 \text{ is } A_i \text{ and } X_2 \text{ is } B_i \text{ then } Y \text{ is } C_i$$

其中， A_i, B_i, C_i 均为论域上的模糊子集，其模糊关系为

$$R_{(x,y)} = \bigcup_{i=1}^n (A_i \times B_i) \times C_i$$

在实际系统中，一般选取误差信号 E （或 e ）和误差变化信号 EC （或 ec ）作为模糊控制器的输入变量 X_1 和 X_2 ，而把受控变量的变化作为输出变量 Y ，如图7.13所示。

(3) 多输入多输出结构

通常人们对具体事物的逻辑思维不超过三维，很难对多输入多输出系统直接提出控制规

则。可以采取模糊解耦方法，将多输入多输出结构化为双输入单输出结构进行设计，例如，可将模糊关系样本数据 $(X_1, X_2, Y_1, Y_2, Y_3)$ 变换为 (X_1, X_2, Y_1) ， (X_1, X_2, Y_2) ， (X_1, X_2, Y_3) 。

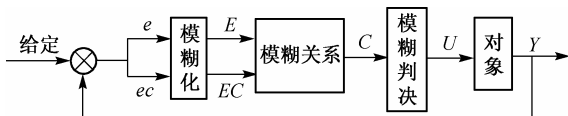


图 7.13 双输入单输出模糊控制系统的结构

3. 模糊控制规则的选取

模糊控制规则是模糊控制器的核心，应考虑下列问题：

(1) 选定描述控制器输入和输出变量的模糊状态

如果用较多的状态来描述每个变量，可以制订比较灵活和精确的规则，但增加了复杂程度。为了兼顾简单性和灵活性，实际应用中通常选取 7 至 9 个模糊状态，即正大（PB）、正中（PU）、正小（PS）、负小（NS）、负中（NM）、负大（NB）和零（ZO）或平均零（AZ）7 个模糊状态；有时还采用正零（PO）和负零（NO）。

(2) 规定模糊集

模糊集表示各模糊状态。在规定模糊集时，必须首先考虑模糊集隶属函数曲线的形状。误差变化范围较大时，采用分辨率较低的模糊子集，使输入误差变化时引起的输出变化比较平缓；而当误差接近零时采用分辨率较高的模糊子集，使输入误差引起的输出变化比较剧烈。例如，通过对数变换的非线性量化映射能够在设定点附近提供较大的控制灵敏度，改善信噪比。分为 8 个模糊状态的模糊集 A 隶属函数赋值示例如表 7.1 所示。

表 7.1 模糊集 A 隶属函数赋值

		-6	-5	-4	-3	-2	-1	-0	+0	+1	+2	+3	+4	+5	+6
A_1	PB	0	0	0	0	0	0	0	0	0	0	0.1	0.4	0.8	1.0
A_2	PM	0	0	0	0	0	0	0	0	0	0.2	0.7	1.0	0.7	0.2
A_3	PS	0	0	0	0	0	0	0	0.3	0.8	1.0	0.5	0.1	0	0
A_4	PO	0	0	0	0	0	0	0	1.0	0.6	0.1	0	0	0	0
A_5	NO	0	0	0	0	0.1	0.6	1.0	0	0	0	0	0	0	0
A_6	NS	0	0	0.1	0.5	1.0	0.8	0.3	0	0	0	0	0	0	0
A_7	NM	0.2	0.7	1.0	0.7	0.2	0	0	0	0	0	0	0	0	0
A_8	NB	1.0	0.8	0.4	0.1	0	0	0	0	0	0	0	0	0	0

(3) 建立模糊控制规则库

建立模糊控制规则库的原则是，对于任意的输入应确保它至少有一个可适用的规则，而且规则的适用程度应大于一定的数，如 0.5。

模糊控制规则的建立选择和确定主要依靠经验和工程知识，通常采用经验归纳法和推理合成法。

① 经验归纳法。建立模糊控制规则的原则是，当误差较大时，控制量应当尽可能快地减小误差；当误差较小时，除了消除误差外，还必须考虑系统的稳定性，以求避免不需要的超调和振荡。根据控制过程中的实际经验,可以写出如表 7.2 所示的模糊控制规则表和相应的模糊条件语句。

例如，对于负误差情况，如果误差较大，为 NB 或 NM，而且误差变化也是负的（NB，NM 或 NS），则选择 PB 作为控制量，以便尽快消除误差，足够快地加速控制响应；如果误差变化为正，表明误差趋于减小，则选择较小的控制量；误差变化为 PS 时，控制量取 PM；而误差变化为 PM 或 PB 时，控制量取 AZ。当误差较小（NS，AZ 或 PS）时，为使控制系统尽快稳定，并防止超调，控制量是根据误差变化确定的。如果误差为 NS，误差变化为 PB，则取控制量为 NS。对于正误差，相应控制量可被对称地确定。

表 7.2 模糊控制规则表

<div> <div>U</div> <div>ΔE</div> </div>		NB	NM	NS	AZ		PS	PM		PB
E	NB	PB			PM			AZ		
	NB	PB			PM			AZ		
	NS	PM			PM		AZ	NS		
	AZ		PM	PS	AZ		NS		NS	
	PS		PS	AZ		NM				NM
	PM	AZ			NM			NB		
	PB	AZ			NM			NB		

② 推理合成法。根据已有的输入输出数据对，通过模糊推理合成，求取模糊控制规则。最小运算推理方法简单，在模糊控制中得到广泛应用，其缺点是推论结果经常不够平滑。两条二输入规则的最小运算推理过程如图7.14所示。

$$\begin{aligned}
 C' &= (A' \text{ and } B') \circ \{ [(A_1 \text{ and } B_1) \rightarrow C_1] \cup [(A_2 \text{ and } B_2) \rightarrow C_2] \} \\
 &= \{ (A' \text{ and } B') \circ [(A_1 \text{ and } B_1) \rightarrow C_1] \} \cup \{ (A' \text{ and } B') \circ [(A_2 \text{ and } B_2) \rightarrow C_2] \} = C'_1 \cup C'_2
 \end{aligned}$$

其隶属函数为

$$\begin{aligned}
 \mu_{C'_1}(y) &= a_{A_1} \wedge a_{B_1} \wedge \mu_{C_1}(y) \\
 \mu_{C'_2}(y) &= a_{A_2} \wedge a_{B_2} \wedge \mu_{C_2}(y)
 \end{aligned}$$

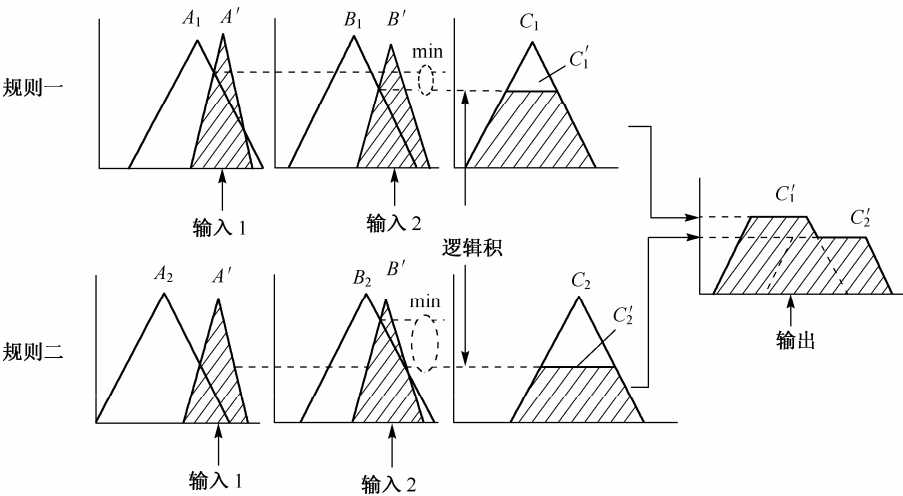


图 7.14 两条二输入规则的最小运算法推理过程

(4) 解模糊

模糊推理的结果是根据各条相关规则推理所得的输出变量隶属函数所组成的图形，称为输出变量的可能性分布，如图 7.14 中两个重叠梯形的外轮廓线。为了从推理所得的模糊控制作用中产生精确的控制作用，需要确定一个值来最好地代表这种分布。常用的方法有：

- ① 最大隶属度平均法 (Mean Of Maximum, MOM)。在输出变量可能性分布曲线上，取隶属度最大的线段的中点，以其横坐标值作为解模糊的输出。
- ② 重心法 (Center Of Gravity, COG)。求输出变量可能性分布曲线所包围面积的重心，以其横坐标值作为解模糊的输出。
- ③ 加权平均法 (Weighted Mean Method, WMM)。输出执行量 u 与加权系数 k_i 有关：

$$u = \sum k_i \cdot y_i / \sum k_i \quad (7.20)$$

灵活选择与调整加权系数 k_i 可以改善系统的响应特性，如果取加权系数 k_i 等于其隶属函数，那就是重心法。

为了增强微机模糊控制的实时性，节约系统开销，可以将离线计算模糊矩阵的结果制成控制表存储。运行时，查表得到确定的输出控制量。

7.1.4 模糊控制设计实例

与一般的工业窑炉相比较，玻璃工业的主要热工设备——玻璃窑炉是一个多变量、动态特性复杂而且控制要求又较高的控制对象。其中，用来生产彩色显像管屏玻璃的屏炉比一般的玻璃制品（如平板玻璃、玻璃器皿等）窑炉对控制系统的要求更高。在屏炉内，原料发生一系列的物理、化学和物理-化学反应，经过熔化、澄清、均化和冷却，最终形成均匀的、符合成型温度要求的熔融玻璃液，整个工艺过程非常复杂。下面通过一个玻璃窑炉控制实例来说明模糊控制策略的应用。

1. 玻璃屏炉工艺过程

屏炉由熔化池、工作池、蓄热室和通道 4 部分组成，其炉体俯视图和纵剖面图如图 7.15 和图 7.16 所示。玻璃配合料由 2 台毯式加料机从 1# 和 2# 投料口加入到熔化池内，在熔化池内经过熔化、除气泡、初步澄清等工艺过程，得到初步澄清的熔融玻璃液。熔融玻璃液由流液洞（起到冷却作用）流入工作池（又称为澄清池或冷却部），在工作池内玻璃液经过进一步澄清、冷却和均化，进入 101 和 102 两条通道，在通道内玻璃液经过严格的温度控制，成为无气泡、均化的、粘度和温度符合成型要求的成品玻璃液，最终由通道口入 11 工位压机压制成屏玻璃。

在整个工艺过程中，燃烧设备采用美国燃烧公司 (CTI) 的天然气燃烧系统进行热工参数控制。大型玻璃窑炉为稳定热工参数，提高窑内温度的均匀性，工艺要求每隔一定时间 (20~30 分钟) 进行一次左右燃烧换向。在换向过程中，会产生很大的扰动因素，从而引起窑内温度、窑压和液面大幅度波动。这是因为换向期间（从换向开始到结束，约 30 秒左右），燃料切断后，窑内温度和窑压大幅度下降，进而引起液面波动；换向结束后，由于窑内温度大幅度下降，为维持工艺需求的温度，需大量增加燃料，由于温度过程的滞后和热容惯性，会带来窑内温度和窑压的再次波动。换向扰动既不利于窑炉经济性燃烧，又降低了窑炉自动控制的精度，必须加以抑制。

换向过程是一个复杂非线性时变过程，很难建立起连续的精确数学模型，应用模糊控制方法对玻璃窑炉换向扰动进行抑制是较好的选择。

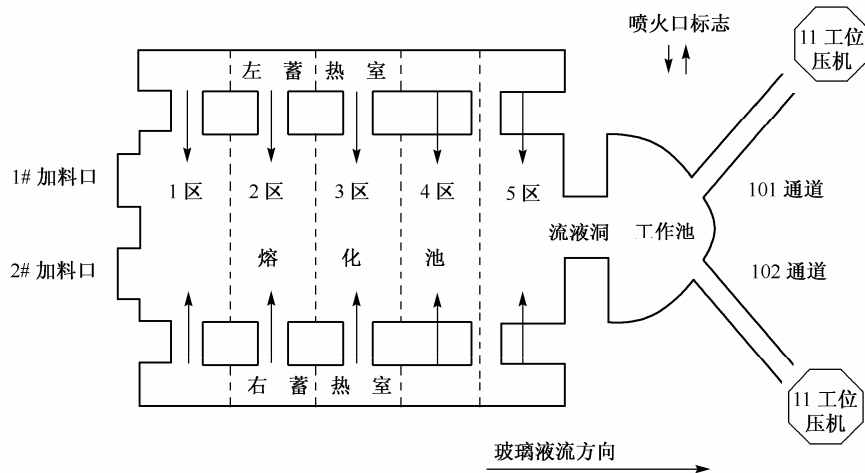


图 7.15 屏炉炉体俯视图

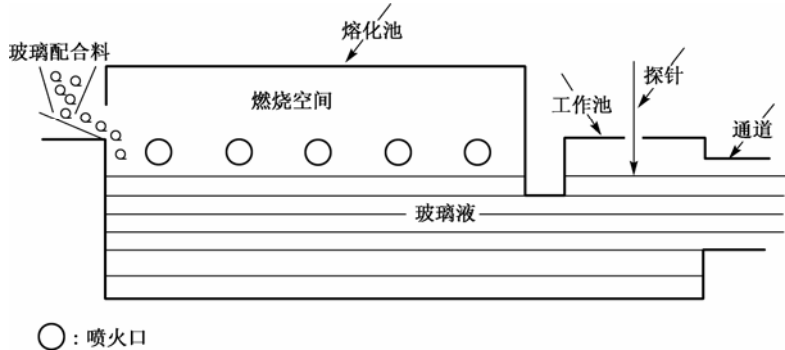


图 7.16 屏炉炉体纵剖面图

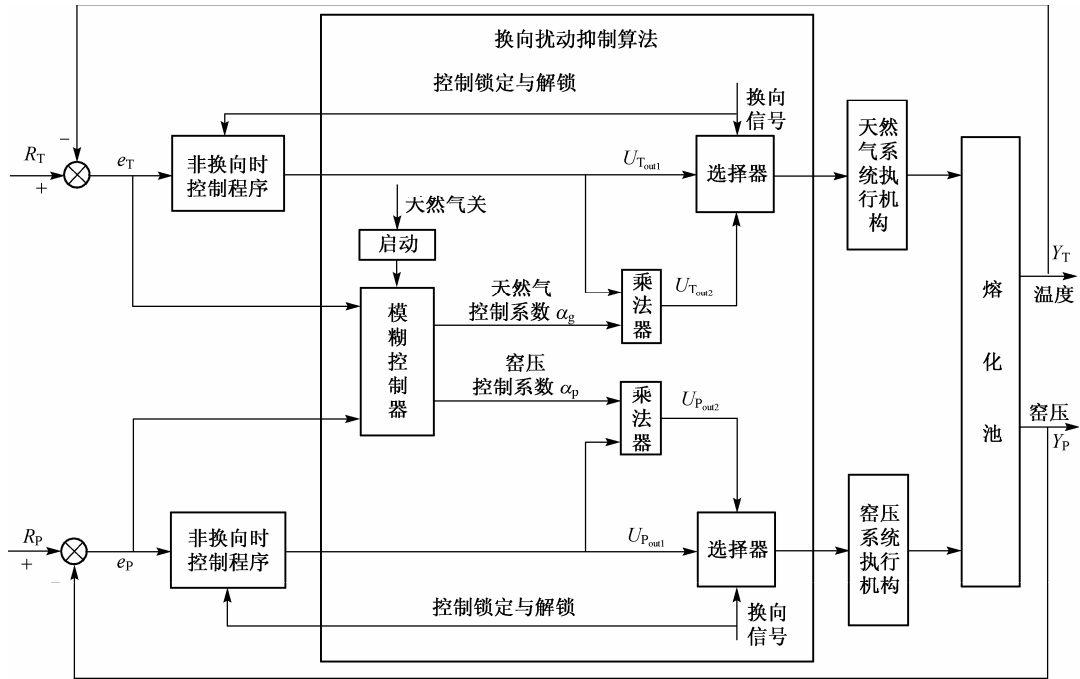
2. 换向扰动抑制算法模糊控制器的设计

模糊控制是建立在模糊逻辑的基础之上，是通过模糊集合来工作的，提供了对现实世界不精确或近似知识的获取方法。经验证明，对一些复杂系统，特别是非线性、时变的、大滞后及带有随机干扰的系统，由于控制对象的数学模型难以建立，模糊控制常优于传统的控制方法。模糊控制对过程参数改变不敏感，具有较好的“鲁棒性”。

换向扰动抑制要综合考虑换向过程中的各种因素，协调控制。在换向预备、换向开始、换向期间、换向结束及换向结束后整个过程中，对燃料、助燃风和烟道抽力等直接影响到窑内温度和窑压的因素，进行协调控制，并给出一系列调节阀位相应动作曲线，以配合换向过程的进行。

根据这种思想，换向扰动抑制算法在换向期间协调助燃风、天然气和炉压调节器的输出值，控制三者的调节器输出改变的时序和改变的幅度。也就是以非换向时间稳定燃烧时调节器的输出为基础，在换向期间将以上三个调节器的输出分别乘以一个 0~1 之间的系数（以下简称“控制系数”）。由于窑炉的非线性与时变因素，采用基于模糊控制的换向扰动抑制算法来实时地得到换向期间的控制系数，使之随着外界的扰动和窑内的热工制度等因素而变化，以提高控制的效果。

天然气和助燃风为配比燃烧，只需得到天然气系统一个控制系数（用 α_g 来表示）即可，窑压的控制系数用 α_p 来表示。加入换向扰动抑制算法后，系统的控制框图如图7.17所示。



说明：选择器在换向期间输出为 $U_{T_{out2}}$ 和 $U_{P_{out2}}$ ，在非换向期间输出为 $U_{T_{out1}}$ 和 $U_{P_{out1}}$ ，用换向信号来启动和停止模糊控制器。

图 7.17 换向扰动抑制控制框图

(1) 确定模糊控制器的输入、输出变量

在这里，选择 α_g 和 α_p 作为控制量，窑压 P 和温度 T 作为被控变量，进而组成双输入、双输出的控制方案。设窑炉熔化池窑压的设定值为 R_p ，实测值为 Y_p ，则窑压偏差为 $e_p = R_p - Y_p$ 。设窑内温度的设定值为 R_T ，实测值为 Y_T ，则温度偏差为 $e_T = R_T - Y_T$ 。在换向期间，窑压是在大幅度地高低波动，故设窑压偏差 e_p 的基本论域是对称的。温度是下降的，控制系数为 $0 \sim 1$ 之间的数，所以温度偏差 e_T 、控制系数 α_g 和 α_p 的基本论域都不对称。故在这里我们选 e_p 的基本论域为 $[-20, 20]$ ， e_T 的基本论域为 $[-10, 30]$ ， α_g 和 α_p 的基本论域为 $[0, 10]$ （将系数都扩大 10 倍，以提高系统的响应速度）。选取基本模糊控制器的输入语言变量为窑内窑压偏差 EP 和温度偏差 ET，输出语言变量为天然气控制系数 CG 和窑压控制系数 CP。下面给出各自的模糊集论域和量化因子：

e_p 对应的模糊集论域为

$$X_{e_p} = \{-6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6\} \quad (7.21)$$

则 e_p 的量化因子 k_{e_p} 定义为
$$k_{e_p} = \frac{n}{e_p} = \frac{6}{20} = 0.3 \quad (7.22)$$

e_T 对应的模糊集论域为

$$X_{e_T} = \{-2, -1, 0, 1, 2, 3, 4, 5, 6\} \quad (7.23)$$

则 e_T 的量化因子 k_{e_T} 为

$$k_{e_T} = \frac{n}{e_T} = \frac{8}{40} = 0.2 \tag{7.24}$$

同理，确定 α_g 和 α_p 的比例因子 k_{α_g} 和 k_{α_p} 的值，对应的模糊集论域为

$$X_{\alpha_g} = X_{\alpha_p} = \{0, \quad 1, \quad 2, \quad 3, \quad 4, \quad 5, \quad 6\} \tag{7.25}$$

则比例因子 k_{α_g} 和 k_{α_p} 的值为

$$k_{\alpha_g} = k_{\alpha_p} = \frac{10}{6} = 1.67 \tag{7.26}$$

对各变量通过各自的量化和比例因子量化到各自的模糊论域上后，确定各自的语言变量值。为使模糊子集对论域的覆盖程度较好，对温度偏差、窑压偏差、天然气控制系数和窑压控制系数的语言变量 EP，ET，CG 和 CP 分别选择了 L ， M ， N 和 Q 的语言变量值如下：

$$L = \{PB, PM, PS, O, NS, NM, NB\} \tag{7.27}$$

$$M = \{PB, PM, PS, O, NS\} \tag{7.28}$$

$$N = \{PB, PM, PS, O\} \tag{7.29}$$

$$Q = \{PB, PM, PS, O\} \tag{7.30}$$

为了在得到模糊控制规则表时方便叙述，对于语言变量 CG 和 CP 的语言变量值，用以下与式(7.29)和式(7.30)对应的词集 Y 和 Z 来表示：

$$Y = \{Y_4, Y_3, Y_2, Y_1\} \tag{7.31}$$

$$Z = \{Z_4, Z_3, Z_2, Z_1\} \tag{7.32}$$

这样，每个语言值就成为各自量化论域式(7.21)、式(7.23)和式(7.25)上的一个模糊子集，这些模糊子集可以由各自的隶属函数来完全刻划。工业过程控制中的模糊概念用正态型模糊变量来描述是比较适宜的。可以用正态分布函数做模糊子集的隶属函数，来确定离散论域中的元素属于各模糊子集的隶属度。对语言变量 EP，ET，CG 和 CP 等就得到在论域式(7.21)、式(7.23)和式(7.25)上以正态分布密度函数为隶属函数时的赋值，将它们组合成表格形式，便得到语言变量 EP，ET，CG 和 CP 的赋值表。

(2) 建立模糊控制器的控制规则
 采用双输入（ e_p 和 e_T ）、双输出（ α_g 和 α_p ）的模糊控制，与之对应的控制规则为

$$\text{IF } EP' = EP_i \quad \text{AND} \quad ET' = ET_j \quad \text{THEN} \quad CG' = CG_k \quad \text{AND} \quad CP' = CP_m \tag{7.33}$$
 根据现场操作人员的成熟经验和作者对窑炉换向过程的详细观察，同时结合式(7.33)，便形成了模糊控制规则表，如表 7.3 所示。

表 7.3 模糊控制规则表

CG' CP'		EP'						
		NB	NM	NS	ZO	PS	PM	PB
ET'	NS	$Y_1 \quad Z_3$	$Y_1 \quad Z_3$	$Y_1 \quad Z_3$	$Y_1 \quad Z_2$	$Y_1 \quad Z_2$	$Y_1 \quad Z_1$	$Y_1 \quad Z_1$
	ZO	$Y_1 \quad Z_3$	$Y_1 \quad Z_3$	$Y_1 \quad Z_3$	$Y_1 \quad Z_2$	$Y_2 \quad Z_2$	$Y_2 \quad Z_1$	$Y_2 \quad Z_1$
	PS	$Y_2 \quad Z_3$	$Y_2 \quad Z_3$	$Y_2 \quad Z_3$	$Y_2 \quad Z_2$	$Y_2 \quad Z_2$	$Y_2 \quad Z_1$	$Y_2 \quad Z_1$
	PM	$Y_3 \quad Z_4$	$Y_3 \quad Z_3$	$Y_3 \quad Z_3$	$Y_3 \quad Z_2$	$Y_3 \quad Z_2$	$Y_3 \quad Z_1$	$Y_3 \quad Z_1$
	PB	$Y_4 \quad Z_4$	$Y_4 \quad Z_4$	$Y_4 \quad Z_4$	$Y_4 \quad Z_2$	$Y_4 \quad Z_2$	$Y_4 \quad Z_2$	$Y_4 \quad Z_1$

式(7.33)的控制策略实际是两套，可以归入式(7.34)和式(7.35)：

$$\text{IF } EP' = EP_i \quad \text{AND} \quad ET' = ET_j \quad \text{THEN} \quad CG' = CG_k \quad (7.34)$$

$$\text{IF } EP' = EP_i \quad \text{AND} \quad ET' = ET_j \quad \text{THEN} \quad CP' = CP_m \quad (7.35)$$

式(7.34)的模糊关系 \tilde{R}_{ij} 用于决定天然气控制系数 α_g ，式(7.35)的模糊关系 \tilde{R}_{2ij} 用于决定窑压的控制系数 α_p 。求得 \tilde{R}_{1ij} 和 \tilde{R}_{2ij} 为

$$\tilde{R}_{1ij} = (EP_i \times ET_j)^T \times CG_{ij}, \quad \tilde{R}_{2ij} = (EP_i \times ET_j)^T \times CP_{ij} \quad (7.36)$$

则模糊控制的总模糊关系 \tilde{R}_1 和 \tilde{R}_2 为

$$\tilde{R}_1 = \bigcup_{ij} \tilde{R}_{1ij} \quad \text{和} \quad \tilde{R}_2 = \bigcup_{ij} \tilde{R}_{2ij} \quad (7.37)$$

当得到了两个模糊输入 EP' 和 ET' 时，就可以由模糊推理合成规则，得到控制量 CG' 和 CP' ：

$$CG' = (EP' \times ET')^T \circ \tilde{R}_1, \quad CP' = (EP' \times ET')^T \circ \tilde{R}_2 \quad (7.38)$$

(3) 模糊判决（解模糊）

模糊控制器的输出 CG' 和 CP' 是模糊子集，不能直接控制被控对象，在这里选用最大隶属度法来将模糊量还原为精确控制量。

这样，我们由式(7.38)得到 CG' 和 CP' 后，通过式(7.26)所定义的比例因子 k_{α_g} 和 k_{α_p} 将模糊量还原为精确控制量 α_g 和 α_p ：

$$\alpha_g = k_{\alpha_g} \times n_1 \quad \text{和} \quad \alpha_p = k_{\alpha_p} \times n_2 \quad (7.39)$$

其中， n_1 和 n_2 为模糊集合 CG' 和 CP' 通过最大隶属度法在式(7.26)所定义的论域上进行量化后得到的元素。这样，通过离线计算可得 CP 和 CG 的控制查询表，将它们存储在计算机中。在控制时，把经过量测和变化得到的 EP 和 ET 值直接与控制查询表中的行和列比较，即可得到所需的控制系数语言变量 CP 和 CG 的值，然后通过式(7.39)计算出精确的控制系数值。

3. 现场检测结果及分析

图 7.18 和图 7.19 是某大型彩色玻璃厂技术改造前玻璃窑炉换向燃烧典型的窑内温度和窑压波动曲线图。其中，工艺温度设定为 1606℃，窑压设定为 28 Pa。在换向期间，窑内温度下降了 30℃，窑压波动约为 ±10 Pa 左右；并且在换向结束后，温度出现了很大的超调。这是因为，在换向期间温度大幅度下降，换向结束后必须加大燃料量，由于温度的滞后和窑炉的热容惯性，温度一定会有超调。因此，换向过程严重地打破了窑内的热平衡，给自动控制带来了严重的影响。

按照图 7.17 所示的控制框图进行换向扰动抑制控制系统的现场投运，在非换向期间，模糊控制器锁定，选择器输出为 $U_{T_{out1}}$ 和 $U_{P_{out1}}$ ；换向开始后，非换向控制程序输出锁定，同时选择器输出为 $U_{T_{out2}}$ 和 $U_{P_{out2}}$ 。当关天然气的信号到来之后，启动模糊控制器，使得窑压、天然气和助燃风的控制输出由通过模糊控制器算得的控制系数来决定，保证各阀门的开度按照一定的曲线变化，从而抑制换向期间的扰动。当换向结束后按照一定的解锁顺序进行解锁时，同时锁定模糊控制器，由非换向时控制程序来进行控制。

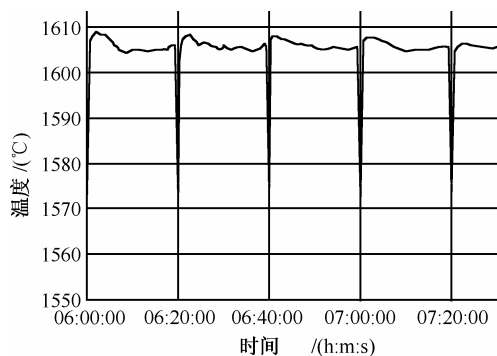


图 7.18 换向燃烧窑内典型温度波动曲线图

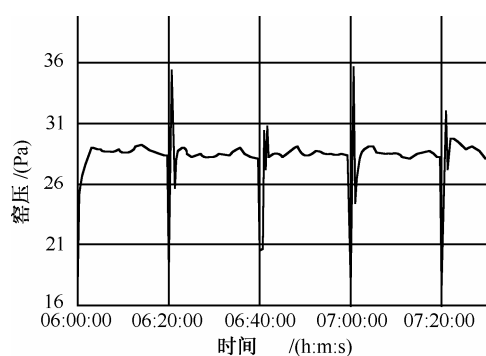


图 7.19 换向燃烧窑内典型窑压波动曲线图

图7.20和图7.21是经过技术改造,加入换向扰动抑制算法后,换向燃烧窑内温度和窑压曲线图。由图可见,窑内的温度下降最多为 14°C 左右,窑压的波动幅度为 $\pm 3\text{ Pa}$ 。与未加入换向扰动抑制算法之前的温度和窑压波动曲线图7.18和图7.19相比,在换向期间,加入了换向扰动抑制算法后,温度下降的幅度得到了很好的遏制,所以,当换向结束后,燃料相对非换向期间的变化量就较小,温度超调很小,温度能很快稳定在设定工作点上。由于各阀门的开度按照一定的曲线变化,燃料和助燃风等逐渐进入窑内,使得窑压波动也得到了很好的控制,较未加入换向扰动抑制算法之前要稳定得多。因此,扰动抑制算法的加入,减小了换向过程对窑内温度和压力的冲击,保证了自动控制的精度和稳定性。

大型玻璃窑炉的火焰换向是一个复杂非线性、时变过程,针对换向对窑炉热工参数的影响,在窑炉换向过程中加入基于模糊控制的换向扰动抑制算法,来优化控制系数 α_g 和 α_p ,保证了换向期间天然气流量控制阀阀位、助燃风流量控制阀阀位和窑压引风机转速按照一定的曲线进行输出。有效地抑制了换向期间窑内窑压和温度的波动,保证了窑内温度和窑压等热工参数的稳定,从根本上保证了成品玻璃液的质量。

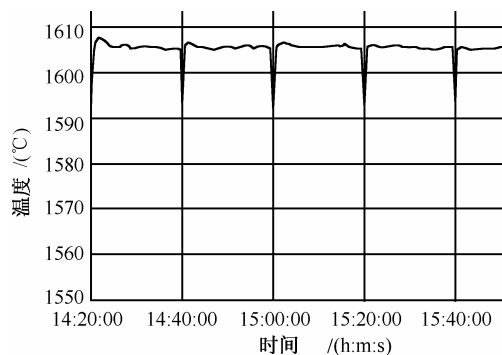


图 7.20 加入换向扰动抑制算法后温度曲线图

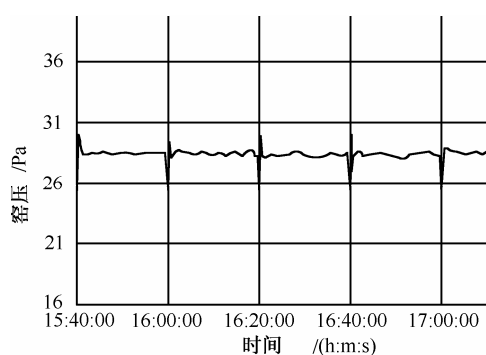


图 7.21 加入换向扰动抑制算法后窑压曲线图

7.2 专家控制

7.2.1 专家系统

专家是某一领域内解决专门问题的高手,他们不但掌握了处理问题的详细专业知识,并

且具有丰富的经验，能够组合应用启发知识和多种经验，高效、准确地做出推理、判断、决策和结论。

专家系统是模拟人类专家解决专门问题的计算机程序系统，能够利用该领域专家水平的大量的专门知识与经验，进行推理、判断和决策，解决复杂困难的问题。

专家系统的基本功能取决于它所含有的知识，因此也把专家系统称为基于知识的系统（Knowledge-Based System）。

1. 专家系统的特点

(1) 启发性

专家系统所要解决的问题一般没有算法解，并且往往要在不精确、不确定或不完全的信息基础上进行推理，求解过程也无法或很难用简单的数据流或精确的逻辑判断进行精确的描述，其问题求解（Problem-Solving）知识不仅包括理论知识和常识，而且包括专家本人可能是不完全的和不准确的启发知识。

与常规程序进行数据处理和数字计算不同，专家系统强调符号处理和符号操作（运算），使用符号表示知识、概念，进行推理。

(2) 透明性

专家系统能够应用知识库内的知识和问题求解过程中产生的中间结果，解释本身的推理过程和回答用户提出的问题，以便提供一个透明的界面，让用户了解、信任、接受推理过程。解释推理路径也可以验证求解过程的合理性，知识工程师和用户能够从中发现推理失败之处，为改进系统的求解结果获得经验。

(3) 灵活性

完善的专家系统具有自学习的能力，可以不断扩充和完善知识库，修改与更新原有知识，进而不断改善系统性能。另一方面，与知识库相对独立的推理机，能够从知识库内选择各种相关知识，并根据具体求解问题的特点构造出问题求解序列。推理机的程序与知识库的具体结构和组成无关，对知识库的修改和扩充无须改动推理机。

2. 专家系统的结构

专家系统的基本结构如图7.22所示，主要包括：

(1) 人机接口

人机接口是用户与系统方便直观地进行信息交流的媒介，其功能是识别与解释用户向系统提供的命令、问题和数据等信息，并把这些信息转化为系统的内部表示形式；另一方面，人机接口也将系统向用户提出的问题、结果和解释以用户易于理解的形式提供给用户。

(2) 数据库

用于存放需要的原始数据和系统运行过程中产生的信息，包括原始信息、推理的中间结果、推理过程的记录。数据库中，问题求解状态下的符号或数据的集合统称为事实，可以而且也是经常变化的。

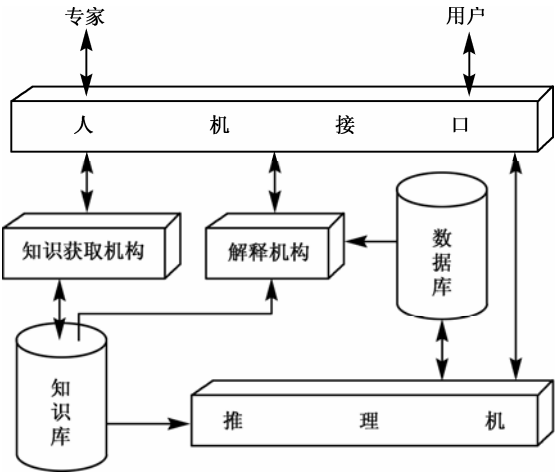


图 7.22 专家系统结构图

(3) 知识库

用于存放问题求解需要的具有规律性、普遍性的整个专业领域的知识，是相对稳定的。包括与领域问题有关的一般知识，以及描述各种事实的知识，如与该领域有关的定义、定理和确定的或不确定的推理法则等，也包括作为专家经验的判断性知识、启发式知识。

知识库的建造包括知识获取及知识表示。知识表示的核心是选择合适的数据结构，把所获取的专家知识进行形式化处理并存入知识库中。

(4) 推理机

在一定的控制策略下，根据数据库的当前状态，按照类似专家水平的问题求解方法，调用知识库中与当前问题有关的知识进行分析、判断和决策，推出新的事实或者执行某个操作。对于复杂问题，能根据问题求解的情况随时调整推理的策略。

(5) 知识获取机构

负责建立、修改与扩充知识库，实现专家系统的自学习，并维护知识库的一致性、完整性。知识获取机构具备知识变换手段，能够通过与专家的对话，增加或修改知识库中的知识，也能通过系统求解过程中自动积累和用户对每次求解的反馈信息（实时控制中还包括实时检测到的有关信息），自动进行知识库的修改和完善。

(6) 解释机构

根据知识库和数据库中对推理过程的记录，对求解过程进行说明，并回答用户提出的问题。在故障诊断、生产操作指导等实时专家系统中，解释机构更是重要的输出方式。

3. 实时专家系统

实时专家系统是满足实时性要求的专家系统，即在系统所要求的时间期限内，能够完成相应的推理过程，满足专家系统功能的要求。

(1) 实时专家系统的特点

- ① 输入信息主要来自监视外界环境的多个独立传感器，自动循环，自动采集；
- ② 输出信息直接送往过程的控制器及向生产现场送出诊断、预报、操作指导等信息；
- ③ 信息量较大，是连续时变的实时数据，可实时响应、处理发生的异步事件。

(2) 实时专家系统的类型

① 专家控制系统

专家控制系统应用专家系统技术，实时地从外界获取被控系统的当前状态，进行预定的处理和推理，并根据结果对外界被控系统进行及时的控制，达到预定要求。

② 非直接控制系统

- 专家操作指导系统。将计算机构成的专家系统在记忆与计算、演绎推理与匹配搜索上的时空优势和人的直觉、顿悟等创造性思维的智能优势相结合，形成既快速精确，又敏锐灵巧的人机协同思维。实时向操作人员报告系统的运行和控制情况，对生产过程进行监督、干预，并给出操作提示。
- 专家故障诊断系统。充分利用专家的经验、直觉等浅层知识和数学模型深层知识，及时完整地进行故障检测和诊断，检测故障的发生并报警，确定故障的部位、原因、程度及影响，提供消除故障的措施，根据对当前和历史状况的分析，进行事故和险情的预测预报。
- 专家信息处理系统。分析现场信息，在数据输入后的较短时间内完成推理。

7.2.2 专家控制系统

专家控制系统是应用专家系统概念和技术，模拟人类专家的控制知识与经验而建造的控制系统。

许多实际过程，具有强烈的非线性、时变性及不确定性，无法获得精确数学模型，但工程技术专家却能凭经验对它们进行很有效的控制，而基于单纯数学解析体系的传统控制理论很难处理对象或过程中的一些定性信息，也很难运用人的经验、知识、技巧和直觉推理，因而难以满足对复杂的未精确建模系统的控制要求。

专家控制把经验与控制算法结合起来，即把符号推理与数值运算结合起来，符号推理在某种意义上代替了人类的操作。

1. 知识的表示

在专家控制系统中，可以认为知识是由特定领域的描述、关系和过程关联组成的信息结构。

(1) 知识的分类

按知识在专家控制系统中的作用及表示，可有如下分类。

① 事实性知识。描述所论领域内的有关概念、事实、事物的属性、系统的状态、环境和条件等。一般采用直接表示的形式。例如，“速度控制系统已处于稳速运行，其速度误差小于0.5%”，“反应器的温度是350℃”等。

② 过程性知识。描述完成某项任务的过程，系统状态的变化，问题求解的操作等。一般都是一些规律性的知识，由问题领域内的规则、定律、定理及经验构成，其表示形式可能是一个标准函数、一套解决某个问题的标准子程序或是一组控制规则。

③ 控制性知识。是关于知识内容、特性和运用的知识，也称为元知识、超知识。具体可分

- 刻划领域知识内容和结构的一般特性的元知识，如知识的产生背景、范围、可信程度等；
- 运用知识的元知识，如在推理机作推理时所用到的推理策略、搜索策略、限制策略等。

(2) 知识的表示方法

为了对知识进行符号化的描述和组织，需要一种计算机能够接受的数据结构。在专家控制系统中，选择知识表示方法有以下考虑：

① 正确有效的表达能力，首先是能充分地表示领域知识。例如，产生式表示法适于表示具有经验性、因果性特点的过程控制知识。将框架表示与产生式表示法结合起来，比较容易反映既复杂而又存在隶属关系的系统。该系统有多个部件，部件又由多个子部件组成，部件与子部件间既有相同属性又有不同属性的结构关系。

② 符合人类的思维习惯，易于理解。

③ 便于进行推理，能够有效地利用所表达的知识。

④ 能够方便灵活地对知识进行修改、删除和扩充。

(3) 产生式规则

产生式规则表示法是在专家控制系统中采用最多的知识表示法。采用产生式规则表示法的专家控制系统简称为产生式系统。

产生式规则的形式是，“如果条件成立，则进行操作”，其一般表示式为

R#: IF <条件> THEN <操作>

其中，R#为规则在规则库中的序号。

<条件>部分也称为前提、前项或产生式左边，可以是多个因素的逻辑组合。例如：

IF (<条件 1> OR <条件 2>) AND <条件 3> ... THEN <操作>

<操作>部分也称为结论、后项或产生式右边，也可以是一个具有 IF ... THEN ... 形式的子规则集，即产生式规则可以嵌套。例如：

IF <条件 1> AND <条件 2>

THEN (IF <条件 3> THEN <操作>)

<条件>和<操作>之间的关系可以包括解析表达式、模糊关系、因果关系和经验规则等多种形式或它们的结合。例如：

IF $x \in W$ THEN $u(k) = u(k-1)$

可以引入一个与时间有关的因子或时间函数 $T(t)$ ，并采用产生式规则的形式来表达与时间有关的动态知识。例如：

IF $T(t)$ >规定的时间限度 THEN <操作>

2. 产生式系统的推理

(1) 推理过程

产生式系统的推理过程可以分为如下三个阶段。

① 模式匹配。将当前数据库中的事实与知识库中各规则的条件部分相比对，两者完全匹配或近似匹配的规则称为触发规则。将所有的触发规则取出，组成触发规则集。

② 选出启用规则。从触发规则集中按某种策略选出一条执行的规则，称为启用规则。

③ 执行操作。执行启用规则的操作部分，输出其结果并更新数据库。

产生式系统的推理就是“匹配—选用规则—操作”的周期循环，使得数据库的内容更新，下一周期可以选择不同的启用规则。这样反复循环，最终使得数据库的内容包含目标状态，即问题得到求解。

(2) 竞争消解策略

从触发规则集中按某种策略选出一条启用规则称为竞争消解。常用的策略有

- 根据实际问题的特征来定义优先权，选取优先权最高的规则；
- 将所有规则进行线性排序，选取最先匹配成功的规则；
- 将规则按执行过的次序排序，选取最近执行过的规则；
- 选取条件描述复杂性高的规则；
- 选取原来没有用过的新规则；
- 规定一条规则只可执行一次（Agenda 法）；
- 用随机数发生来决定启用规则的序号；
- 检查可用规则是否会产生数据冗余，取冗余小的规则。

(3) 推理方向

产生式系统的推理方向可分为三种：正向推理、反向推理和双向推理。

① 正向推理。也称为事实驱动推理方式。推理方向是由已知的事实出发直到结论。推理机根据原始信息，在知识库中寻找能与之匹配的规则。若找到，则将该规则结论部分作为中间结果，并利用它继续与知识库中的规则匹配，直到得到最终结论，如图7.23所示。

正向推理简单、易于实现，在较简单的专家控制系统中获得广泛的应用。但正向推理目的性不强，往往要用启发性知识来控制中间结论的选取，对复杂的大型专家控制系统困难较大。

② 反向推理。也称为目标驱动方式。反向推理过程如图7.24所示。选定一个假设目标，然后由此出发进一步寻找支持假设的证据，即在知识库中寻找能导出该目标的规则集。若这些规则中某条规则的条件部分与数据库中事实相匹配，则执行该规则，否则将该规则条件部分作为子目标，递归执行上述过程，直到总目标被求解或者不存在能导出目标（或子目标）的规则为止。

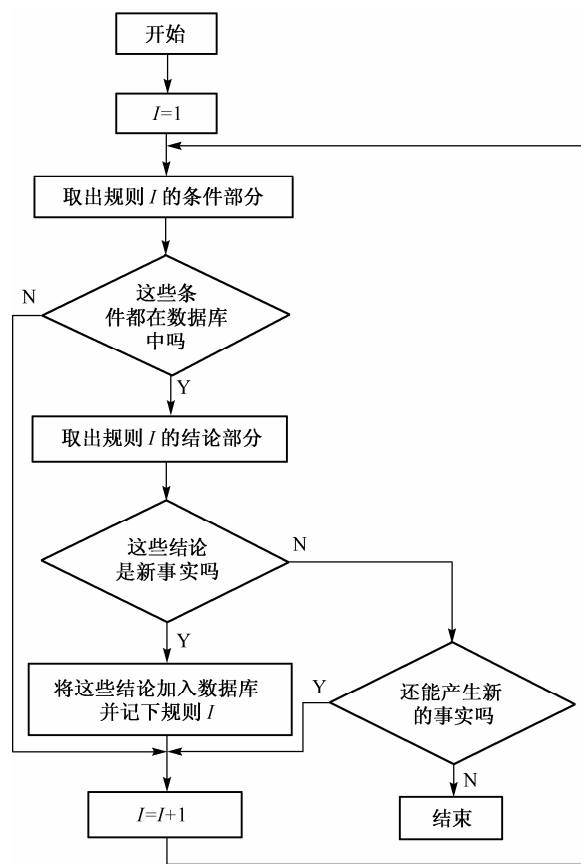


图 7.23 正向推理过程示意图

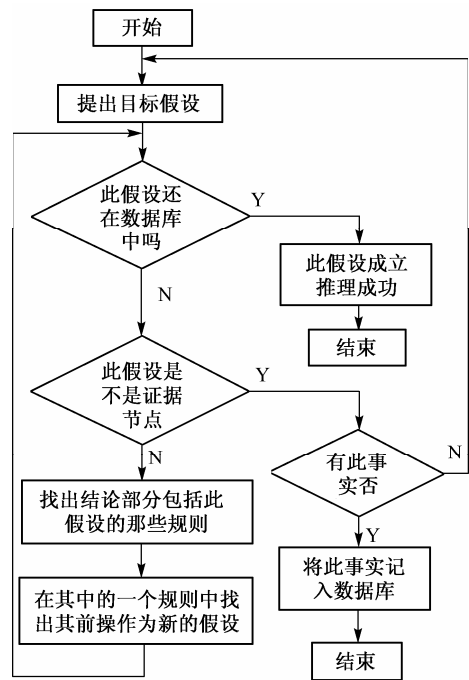


图 7.24 反向推理过程示意图

反向推理在选择初始目标时具有很大的盲目性。初始假设选得准确，问题求解的效率就高，否则会导致许多无用的操作过程。它比较适用于结论单一或直接提出结论要求证实的场合。

③ 双向推理。又称为正反向推理或混合推理，推理过程如图7.25所示。先根据已知事实出发通过正向推理帮助提出假设，再用反向推理进一步寻找支持假设的证据。这样一前一后地反复进行推理，直到得到结论为止。

双向推理具有正反向推理的优点，但双向推理过程中要求正反向推理在某些子目标上接合，因此，接合点判断及正反向比重的均衡就成为双向推理的难点。

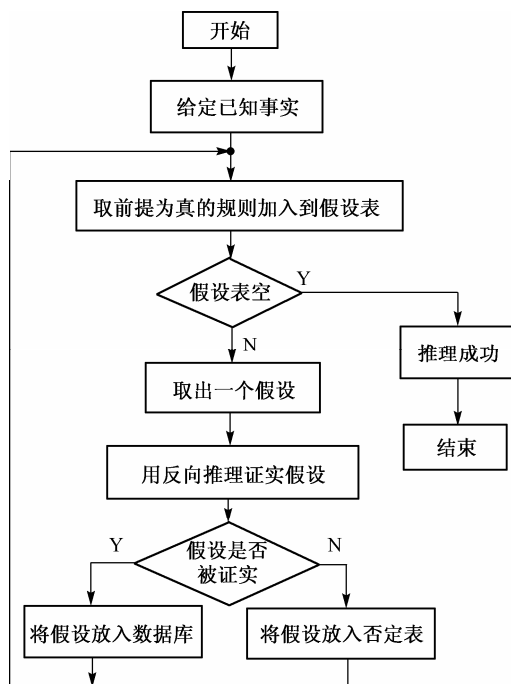


图 7.25 双向推理过程示意图

7.2.3 专家控制系统的设计

1. 模型描述

专家面对的对象及其环境具有非线性、时变性、不确定性及信息不完全性。进行控制的方式也多种多样——精确的、模糊的或灰色的，即当某些信息不明确时，凭直觉、常识和信念进行控制，并在控制过程中再逐步明确起来。所以，在专家控制中，对于被控对象和控制器的描述可采用定量的、定性的、精确的、模糊的及灰色的等各种形式。主要模型有

(1) 解析模型

在常规控制理论中，通常采用数学解析模型。解析模型一定是确定性的，信息论中称为白色的，可以用数学公式来精确表达，例如，微分方程、差分方程、传递函数、状态方程等。

(2) 模糊模型（模糊模型）

在控制过程中，当只掌握对象的一些定性知识时，经常使用模糊描述。

(3) 灰色模型

灰色性即信息不完全性，特点是“部分已知，部分未知”。灰色性是指对象的外延，即符合该对象定义的所有元素的全体是明确的，而对象的内涵，即该对象区别于其他对象的基本属性不明确。

灰色性可用灰色系统理论来处理，例如，灰数的白化处理。通过补充信息，将灰色的内涵转化成白色内涵。

(4) 综合模型

根据不同的情况，用各种知识表达方式通过有机的结合，形成一个综合模型。

2. 专家控制系统结构

专家控制系统结构可分为间接专家控制和直接专家控制。

间接专家控制也称为专家监督控制,专家系统只是通过对常规控制器结构的切换、控制算法的选择、参数的调整,间接地影响被控制过程,其一般结构如图7.26所示。

在直接专家控制系统中,专家系统包含在控制回路中,根据所测到的过程信息及知识库中的规则,导出每一采样时刻的控制信号,其一般结构如图7.27所示。

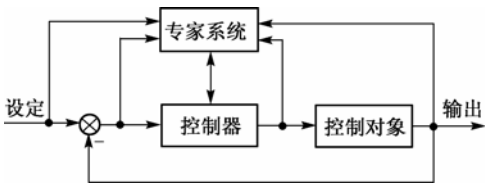


图 7.26 间接专家控制系统

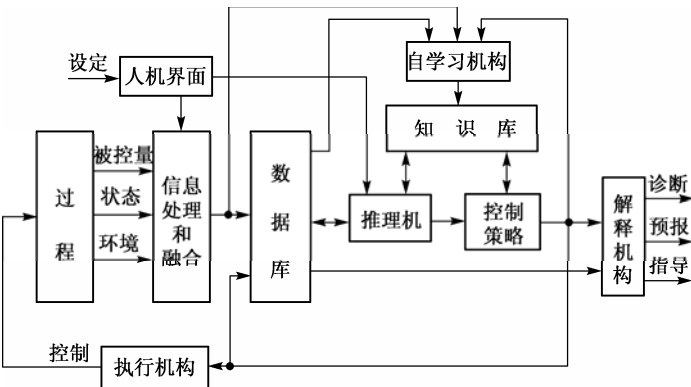


图 7.27 直接专家控制系统

- (1) 信息处理和融合
 - ① 获取实时信息。采用检测仪表、软测量技术、模式识别等手段测量被控变量,观测状态变量,辨识过程环境,并进行去噪声、滤波等预处理。
 - ② 提取特征信息。抽取动态过程的特征信息,识别系统的特征状态,并对特征信息进行必要的加工。例如,计算系统被控量与期望值的误差、误差的导数、误差的第*i*次极值等。
 - ③ 信息融合,采用信息融合(Information Fusion)的有关理论和方法对来自多传感器系统的信息进行综合处理,使传感器彼此间协调工作,有效地利用信息的有机组合蕴含的信息特征,协调成统一的表达方式,可以较完整地完成任务对环境和对象特征的描述,已经在C³I(指挥、控制、通信与情报)等军事和民用的各个领域得到越来越广泛的应用。
- (2) 控制策略

专家控制系统可以采用灵活多变的控制策略:根据不同的对象、不同的动态响应状态、不同的控制要求,采用解析的、模糊的或知识型的等多种模态;开环控制、闭环控制、正反馈控制、负反馈控制、定性决策与定量控制等多种控制模块的结合;能通过在线获取的信息灵活地修改控制策略或控制参数。

- (3) 自学习机构

在符合专家控制系统实时性要求的自学习机构中,有如下较流行的方法。

 - ① 统计学习法。规则成立的概率越大,说明该规则越可靠可信。在系统运行过程中,不断测试每条规则的可靠程度,并删除成立概率低于某一限度的规则,使知识变得更加灵活可靠,保证系统处于最佳运行状态。

② 奖罚学习法。设置一个判别函数，以加强对过程进行有利的控制作用，反之，则减小。

例如，判别函数取为 $C(k) = e(k) \cdot e'(k)$ ，若当前 $C(k) < 0$ ，说明被控制量能减小偏差，则对控制量 u 进行奖励，如将控制量乘上一个大于 1 的数，使控制量增加；若 $C(k) > 0$ ，则对控制量进行惩罚，减小控制量 u 。

(4) 知识库

缩小规则库的搜索空间有利于搜索的进行，加快推理过程。常用的方法有

① 逐步优化。将知识库分成几个层次来建立，处在低层的知识层较简单，采用较简单的算法和较少的规则。越高层的知识层越复杂，采用较复杂的算法和较多的规则。推理机构先在最低层推理，可以很快地得到一个结果，虽然该结果并非最优。如果时间允许，则进入上一层搜索，以期获得一个更优的结果。

② 产生式规则与框架结合。可以是在规则中包含框架，即将规则的一些项（条件、结论）设计成框架，如果以某种方式使这些框架匹配成功，则认为该规则成立。也可以是在框架中包含规则。把框架的某些槽作为一些规则，以使用框架表达知识，而用嵌套的规则表达细节。

7.3 人工神经网络控制

现代计算机有着强大的计算和信息处理功能，但缺乏向环境学习、适应环境、解决复杂问题的能力，只能机械地执行事先编好的程序。

20 世纪初，人们已经知道人脑是由极大量的基本单元——生物神经元（如图 7.28 所示），

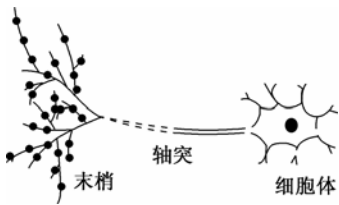


图 7.28 生物神经元

经过复杂的互相连接而成的高度复杂、非线性、并行处理的信息处理系统。虽然单个神经元的反应速度只是毫秒级，但由于人脑的神经元数量巨大（约为 10^{10} 个），每个神经元可与几千个其他神经元连接（总连接数约为 6×10^{13} ），对有些问题的处理速度反而比计算机快得多，每一运算的能耗也比计算机小得多。

人工神经网络（Artificial Neural Networks, ANN）系统就是为模仿人脑工作方式而设计的一种具有大量连接，可以通过学习获取知识并解决问题的并行分布式处理器，它可用电子元件或光电元件实现，也可用软件在常规计算机上仿真。

在人工神经网络中，知识不是像常规计算机那样按地址存在特定的存储单元中，也不像在产生式系统中那样独立表示为每一规则，而是将某一问题的若干知识以连接权的形式分布存储（对应于生物神经元的突触），表示在同一网络中。学习也不是根据一般代码转换成编译程序，而是通过机器学习完成的。神经网络具有很强的学习能力，机器学习程序可以从面向专家、知识工程师和用户的学习范例中提取有关知识，并通过权矩阵及系统参数进行学习训练，将有关知识以网络或动力系统形式表示。多层神经网络的内部可用分块邻接权矩阵和阈值矢量描述。

具有自学习、自适应和自组织，函数逼近和大规模并行处理，信息的分布存储，连续时间的非线性动力学，高度的容错性、鲁棒性和实时处理等特性的人工神经网络，开辟了智能控制系统的新领域。对控制来说，人工神经网络的下列特性特别重要：

(1) 并行分布处理

神经网络具有高度的并行结构和并行实现能力，因而有较好的容错性、鲁棒性和较快的总体处理能力，这特别适于实时控制和动态控制。

(2) 非线性映射

神经网络具有源于其近似任意非线性映射（变换）能力的非线性特性，给非线性控制问题带来了新曙光。

(3) 通过训练进行学习

通过研究系统过去的数据记录进行训练的神经网络，在经过适当训练后，具有归纳全部数据的能力，能够解决用数学模型或描述规则难以处理的控制过程问题。

(4) 适应与集成

神经网络能够适应在线运行，并能同时进行定量和定性操作，可以同时输入大量不同的控制信号，解决输入信息间的互补和冗余问题，并实现信息集成和融合处理。这些特性特别适于复杂、大规模和多变量系统的控制。

(5) 硬件实现

神经网络不仅能够通过软件而且可借助硬件实现并行处理。近年来，由一些超大规模集成电路实现的硬件已经成为商品。这使得神经网络成为具有快速和大规模处理能力的网络。

目前，人工神经网络的应用主要集中在待处理的问题难以用算法来描述，但存在大量可供学习范例的几个方面。例如，信号处理与模式识别、系统辨识和优化、知识工程或专家系统。

在控制领域，已经做出许多努力，把神经网络用于运动过程控制系统、处理控制系统的非线性和不确定性，以及逼近系统的辨识函数等。根据控制系统的结构，可把神经控制的应用研究分为几种主要方法，如监督式控制、逆控制、神经自适应控制和预测控制等。

7.3.1 人工神经网络的基本模型

人工神经网络的结构由基本处理单元及其互连方法决定。

1. 感知器

(1) 神经元及其特性

模拟生物中的神经网络，人工神经网络也是由基本处理单元组成网络的分布并行处理系统；这种基本处理单元也称为神经元，包括了加权和、阈值、非线性映射等运算处理，如图7.29所示。

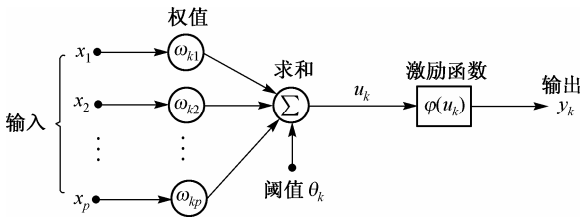


图 7.29 神经元模型

- ① 一组连接权 $\omega_{k1}, \omega_{k2}, \dots, \omega_{kp}$ ，连接强度由各连接上的权值表示，权值为正表示激励，为负表示抑制；
- ② 求和单元，用于求取各输入信息的加权和（线性组合）；
- ③ 非线性激励函数 $\varphi(\cdot)$ ，起非线性映射作用并将神经元输出幅度限制在一定的范围之内，例如， $[0, 1]$ 或 $[-1, +1]$ 之间。通常采用符号函数、分段线性函数或 s 形函数（例如，Sigmoid

函数或双曲正切函数)；

④ 阈值 θ_k 。

对于输入信号 x_1, x_2, \dots, x_p ，神经元 k 的输出 y_k 可以用数学式表达为

$$u_k = \sum_{j=1}^p \omega_{kj} x_j - \theta_k$$

$$y_k = \varphi(u_k) \quad (7.40)$$

(2) 感知器及其学习算法

只有一层计算单元的网络常称为感知器 (Perceptron)，用以实现线性可分函数。对某个具体函数来说，可以采用监督学习法，即根据目标输出 $d(n)$ 与实际输出 $y(n)$ 之间的误差，调整各权系数值的大小来进行学习。当输入样本来自线性可分模式时，上述学习算法在有限步内收敛，所得权值能对所有样本正确分类。

感知器没有隐含节点层，学习过程比较简单，但可以较清楚地揭示人工神经网络的学习机制。

每一个样本都包括输入向量 $\mathbf{x} = [x_1, x_2, \dots, x_p]$ 和目标输出， m 个样本的目标输出向量 $\mathbf{d} = [d_1, d_2, \dots, d_m]$ ，学习时一般将所有样本分为训练样本 (Training Samples) 和测试样本 (Test Samples) 两部分。

实现步骤为

① 初始化：对各连接上的权值 ω_{ki} 和阈值 θ_k 赋予指定值或随机数，得到初始的权值和阈值 $\omega_{ki}(0)$ 和 $\theta_k(0)$ ；

② 根据式(7.40)和输入向量 $\mathbf{x}(n)$ (步数 $n=1, 2, \dots$)，计算在第 n 步时感知器的实际输出向量 $\mathbf{y}(n)$ ；

③ 根据在第 n 步时目标输出量 $d(n)$ 与感知器实际输出量 $y(n)$ 的差距 $\delta(n)$ ，计算权值和阈值的修正量：

$$\delta(n) = d(n) - y(n)$$

$$\Delta \omega(n) = \eta \delta(n) x(n)$$

$$\Delta \theta(n) = \eta \delta(n) \quad (7.41)$$

式中，学习步长 η 用于控制修正的幅度，通常为小于 1 的正数。

权值和阈值的修正有逐步学习 (Step Learning) 和批次学习 (Batch Learning) 两种方式。

● 逐步学习——每输入一个训练样本，计算权值和阈值的修正量后立即调整；

● 批次学习——在学习完所有的训练样本，即完成一个学习循环 (Learning Cycle) 后，根据所有 m 个训练样本的权值和阈值修正量，依照式(7.42)计算出整体修正量后再调整权值和阈值。

$$\Delta \omega_{ki}(n) = \frac{\sum \omega_{ki}^p(n)}{\sqrt{m}}$$

$$\Delta \theta_k(n) = \frac{\sum \theta_k^p(n)}{\sqrt{m}} \quad (7.42)$$

式中， m 为训练样本总数； p 表示训练样本中的第 p 个。

④ 调节权值和阈值：

$$\begin{aligned}\omega(n+1) &= \omega(n) + \Delta\omega(n) \\ \theta(n+1) &= \theta(n) - \Delta\theta(n)\end{aligned}\tag{7.43}$$

⑤ $n = n + 1$ ，返回步骤②。

⑥ 经多次迭代循环，满足终止条件，学习过程结束。

以可以网络收敛，即误差不再有明显变化为终止条件。实际可执行的终止条件常用以下两种：迭代循环达到指定次数；误差小于指定程度。

例如，感知器的误差程度可用总错误率 E 定义：

$$E = \frac{m_E}{m}\tag{7.44}$$

式中， m_E 是误分类样本个数。

误分类样本是指测试样本中，实际输出向量 \mathbf{y} 中指定的输出单元（例如，输出值最大的输出单元），与目标输出向量 \mathbf{d} 中的指定输出单元不一致的样本。

对某个具体函数来说，可以采用监督学习法，即根据目标输出 $d(n)$ 与实际输出 $y(n)$ 之间的误差来调整各权系数值的大小以实现学习。

【例 7.5】利用图 7.30(a) 所示的两输入一输出单层计算单元构成的感知器，通过 4 步学习修正，就可以找到一条直线，将图 7.30(b) 中所示的 3 个白点和 1 个黑点正确地区分为黑白两类样本。

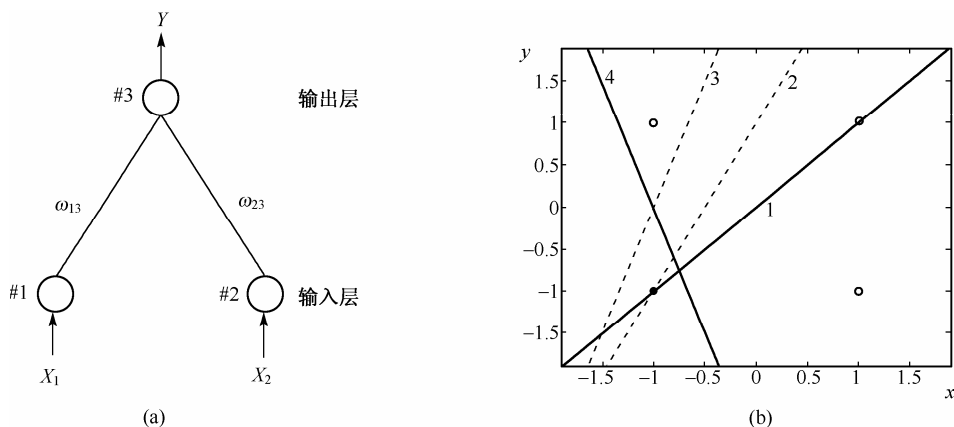


图 7.30 感知器和线性可分的实现

图 7.30(b) 中的 4 点作为感知器的 4 个训练样本，每点的 x, y 坐标构成输入向量 $\mathbf{X} = [X_1, X_2]$ 。黑点的目标输出量 $d=0$ ，白点的目标输出量 $d=1$ ，如表 7.5 所示。

表 7.5 训练样本

样 本 m	X_1	X_2	d
1	-1	-1	0
2	-1	1	1
3	1	-1	1
4	1	1	1

第 n 步时 权向量 $\omega(n)^T = [\omega_{13}(n), \omega_{23}(n)]$

$$v(n)=[\omega_{13}(n)X_1+\omega_{23}(n)X_2-\theta(n)]$$

激励函数 $\varphi(\cdot)$ 采用符号函数, $y=\text{sgn}(v)$

其中
$$\text{sgn}(v)=\begin{cases} +1 & v>0 \\ 0 & v\leq 0 \end{cases}$$

在第 n 步时, 感知器的实际输出为

$$y(n)=\text{sgn}\left[\sum_{j=1}^p\omega_{kj}(n)x_j(n)-\theta_k\right]$$

取学习步长 $\eta=0.5$ 。

① 初始化: 取 $\omega_{13}(0)=1.0, \omega_{23}(0)=-1.0, \theta(0)=0.0$ 。

在图7.30(b)中, 画出

$$\omega_{13}(n)X_1+\omega_{23}(n)X_2-\theta(n)=0$$

即直线 1:
$$x-y=0$$

直线 1 不能将黑点 1 和其他 3 个白点区分。

② 第一步:

对于第一个样本 $X_1=-1, X_2=-1, d=0$,

$$v(n)=[\omega_{13}(0)X_1+\omega_{23}(0)X_2-\theta(0)]=(1.0)(-1)+(-1.0)(-1)-0.0=0.0$$

$$y(0)=0$$

输出误差

$$\delta=d-y=0-0=0$$

权值修正量

$$\Delta\omega_{13}^m=\eta\cdot\delta\cdot X_1=(0.5)(0)(-1)=0$$

$$\Delta\omega_{23}^m=\eta\cdot\delta\cdot X_2=(0.5)(0)(-1)=0$$

阈值修正量

$$\Delta\theta^m=\eta\cdot\delta=(0.5)(0)=0$$

所有四个样本学习过程的第一步如表 7.6 所示。

表 7.6 感知器训练样本学习过程的第一步

m	X_1	X_2	d	v	y	δ	$\Delta\omega_{13}^m$	$\Delta\omega_{23}^m$	$\Delta\theta^m$
1	-1	-1	0	0.0	0	0	0.0	0.0	0.0
2	-1	1	1	-2.0	0	1	-0.5	0.5	0.5
3	1	-1	1	2.0	1	0	0.0	0.0	0.0
4	1	1	1	0.0	0	1	0.5	0.5	0.5

计算所有样本后, 第一步批次学习权值和阈值修正量如下:

$$\Delta\omega_{13}=\frac{\sum_{m=1}^4\Delta\omega_{13}^m}{\sqrt{N}}=\frac{(0.0)+(-0.5)+(0.0)+(0.5)}{\sqrt{4}}=0.0$$

$$\Delta \omega_{23} = \frac{\sum_{m=1}^4 \Delta \omega_{13}^m}{\sqrt{N}} = \frac{(0.0) + (0.5) + (0.0) + (0.5)}{\sqrt{4}} = 0.5$$

$$\Delta \theta = \frac{\sum_{m=1}^4 \Delta \theta^m}{\sqrt{N}} = \frac{(0.0) + (0.5) + (0.0) + (0.5)}{\sqrt{4}} = 0.5$$

更新网络的权值和阈值

$$\begin{aligned}\omega_{13}(1) &= \omega_{13}(0) + \Delta \omega_{13} = (1.0) + (0.0) = 1.0 \\ \omega_{23}(1) &= \omega_{23}(0) + \Delta \omega_{23} = (-1.0) + (0.5) = -0.5 \\ \theta(1) &= \theta(0) - \Delta \theta = (0.0) - (0.5) = -0.5\end{aligned}$$

在图7.30(b)中，画出

$$\omega_{13}(1)X_1 + \omega_{23}(1)X_2 - \theta(1) = 0$$

即直线 2
$$x - 0.5y + 0.5 = 0$$

直线 2 也还不能将黑点和白点区分。

③ $n = n + 1$ ，重复上述计算。各次计算结果如表 7.7 所示。

表 7.7 迭代次数及其计算结果

迭代次数	ω_{13}	ω_{23}	θ
0	1	-1	0
1	1	-0.5	-0.5
2	0.75	-0.25	-0.75
3	0.75	0.25	-0.75

在图7.30(b)中，画出

$$\omega_{13}(3)X_1 + \omega_{23}(3)X_2 - \theta(3) = 0$$

即直线 4
$$0.75x + 0.25y + 0.75 = 0$$

可以看到，直线 4 已经将黑点 1 和其他 3 个白点区分。

2. BP 模型

(1) 前馈型网络与 BP 算法

前馈型网络中，节点分为两类：输入单元和计算单元，各计算单元接受前一层的输入，并输出给下一层，没有反馈。每一计算单元可有任意多个输入，但只有一个输出，该输出可以耦合到下一层任意多个其他节点，作为它们的输入。前馈网络可分为不同的层，输入层和输出层的节点与外界相连，其他中间层则称为隐层。

只有一层计算单元的网络只能实现线性可分函数。反向传播（Back-Propagation, BP）算法解决了增加隐含节点层后学习的困难，促进了多层前馈网络的发展。

如图 7.31 所示，BP 算法的学习过程由正向传播和反向传播组成。在正向传播过程中，输入信息从输入层经隐层单元逐层处理，并传向输出层，每一层神经元的状态只影响下一层神经元的状态。若在输出层不能得到期望的输出，则转入反向传播，将误差信号沿原来的连接通路返回，从后向前修改各层神经元权值，使得误差信号最小。在理论上，BP 网络的节点作用函数只需要是处处可微的，最常用的有 Sigmoid、双曲正切、正弦函数等 S 型函数。

(2) BP 算法学习过程

用于多层前馈网络的 BP 算法学习过程如图 7.32 所示。

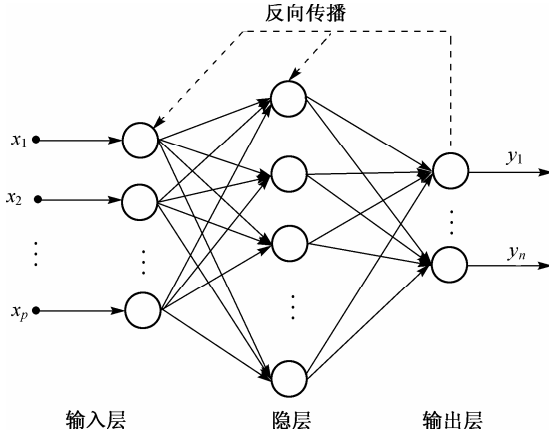


图 7.31 前馈网络与 BP 算法

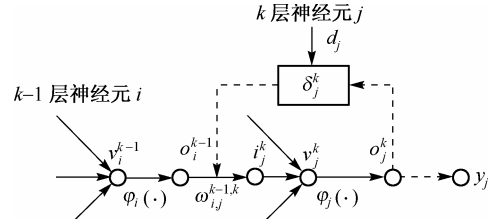


图 7.32 第 $k-1$ 层的第 i 个节点到第 k 层的第 j 个节点的信号流

设 m 层的阶层型网络中，第 k 层的第 j 个节点的输入为 $v_j^k = \sum_i i_j^k$ ，实际输出为 o_j^k ，输入输出函数为 $\varphi_j(\cdot)$ ，即 $o_j^k = \varphi_j(v_j^k)$ 。从第 $k-1$ 层的第 i 个节点到第 k 层的第 j 个节点的连接权为 $\omega_{i,j}^{k-1,k}$ ，即 $i_j^k = \omega_{i,j}^{k-1,k} \cdot o_i^{k-1}$ 。

对于输入模式 P ，如果输出层 m 的第 j 个节点的期望输出为 d_j ，实际输出为 o_j^m 即 y_j ，则输出端均方差为 $E_p = \frac{1}{2} \sum_j (d_j - o_j^m)^2 = \frac{1}{2} \sum_j (d_j - y_j)^2$ 。

为了减小误差 E_p ，根据梯度下降法， $\omega_{i,j}^{k-1,k}$ 的修正量 $\Delta \omega_{i,j}^{k-1,k}$ 为

$$\Delta \omega_{i,j}^{k-1,k} = -\eta \frac{\partial E_p}{\partial \omega_{i,j}^{k-1,k}} \quad (7.45)$$

式中，修正系数 $\eta > 0$ 。

展开式中的偏导数，有

$$\frac{\partial E_p}{\partial \omega_{i,j}^{k-1,k}} = \frac{\partial E_p}{\partial i_j^k} \cdot \frac{\partial i_j^k}{\partial \omega_{i,j}^{k-1,k}} = \frac{\partial E_p}{\partial i_j^k} \cdot o_i^{k-1} \quad (7.46)$$

当 $k = m$ 时， j 是输出层的一个节点，由 E_p 的定义有

$$\frac{\partial E_p}{\partial i_j^k} = \frac{\partial E_p}{\partial o_j^m} \cdot \frac{\partial o_j^m}{\partial i_j^k} = (o_j^m - d_j) \cdot \varphi'(v_j^m) \quad (7.47)$$

当 $k \neq m$ 时， j 是隐含层的一个节点，有

$$\frac{\partial E_p}{\partial i_j^k} = \sum_i \frac{\partial E_p}{\partial i_j^{k+1}} \cdot \frac{\partial i_j^{k+1}}{\partial o_j^k} \cdot \frac{\partial o_j^k}{\partial i_j^k} = \sum_i \frac{\partial E_p}{\partial i_j^{k+1}} \cdot \omega_{j,i}^{k,k+1} \cdot \varphi'(v_j^k) = \varphi'(v_j^k) \cdot \sum_i \frac{\partial E_p}{\partial i_j^{k+1}} \cdot \omega_{j,i}^{k,k+1} \quad (7.48)$$

所以，连接权 $\omega_{i,j}^{k-1,k}$ 的修正公式为

$$\Delta \omega_{i,j}^{k-1,k} = -\eta \cdot \delta_j^k \cdot o_i^{k-1} \quad (7.49)$$

式中,

$$\delta_j^k = \begin{cases} (y_j - d_j) \cdot \varphi'(v_j^m) & k = m \\ \varphi'(v_j^k) \cdot \sum_i \omega_{j,i}^{k,k+1} \cdot \delta_i^{k+1} & k = m-1, m-2, \dots, 2 \end{cases}$$

δ_j^k 可以看做一般化的误差信号, 它的计算是从 $k=m$ 到 $k=2$ 递推进行的。

$\sum_i \omega_{j,i}^{k,k+1} \cdot \delta_i^{k+1}$ 就好像把输出层产生的误差信号 δ_j^m 从输出层开始, 以相反的方向, 加权求和后一层层地传向输入层。误差反向传播算法也因此得名。

在实用中, 为了加快收敛的速度, 减小振动, 可以采用如下的修正公式:

$$\Delta \omega_{i,j}^{k-1,k}(t+1) = -\eta \cdot \delta_j^k \cdot o_i^{k-1} + \alpha \cdot \Delta \omega_{i,j}^{k-1,k}(t) \quad (7.50)$$

式中, α 为指数衰减系数 (正数), t 表示学习的次数。

BP 模型从各个方面都有着重要的意义, 但也存在如下问题:

- 从数学上看, 它是一个非线性优化问题, 这就不可避免地存在局部极小问题, 这也是运用梯度下降法时常常遇到的;
- 学习算法的收敛速度很慢, 通常要几百步、上千步的迭代;
- 网络的隐节点个数选取尚无理论上的指导, 而是根据经验和多次试验来选取的;
- 新加入的样本会影响到已经学完的样本, 刻画每个输入样本特征的数目也要求必须相同。

3. 霍普菲尔德 (Hopfield) 模型

(1) 反馈型网络

反馈型网络中, 所有节点都是计算单元, 它们之间都可互相连接 (一个节点既接受其他节点的输出, 也输出给其他节点), 同时也可接受输入, 并向外界输出 (如图7.33 所示)。

单层全连接的反馈网络引入了反馈, 因而是一个非线性动力学系统, 可以从不同方面利用其复杂的性质, 以完成各种计算功能。

(2) 霍普菲尔德网络

4 个神经元的霍普菲尔德网络如图7.34所示, 其作用过程表示为

$$v_j = \sum_{i=1}^N \omega_{ji} s_i - \theta_j \quad (7.51)$$

$$s_j = \text{sgn}[v_j] = \begin{cases} +1, & v_j > 0 \\ -1, & v_j < 0 \end{cases} \quad (7.52)$$

一般规定 $v_j = 0$ 时, 神经元保持原来状态不变, 可以用下列方程式描述状态变化:

$$s_j(t+1) = \text{sgn} \left[\sum_{i=1}^N \omega_{ji} s_i(t) - \theta_j \right] \quad (7.53)$$

式中, $s_j(t)$ 为任一时刻单元 j 的状态, 整个网络 (包括 N 个单元) 的状态记为

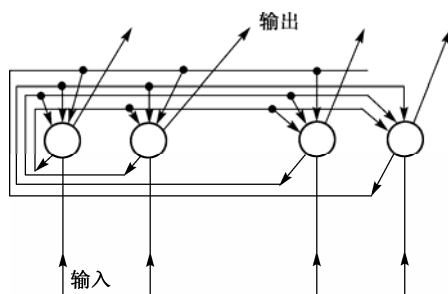


图 7.33 单层全连接的反馈型网络

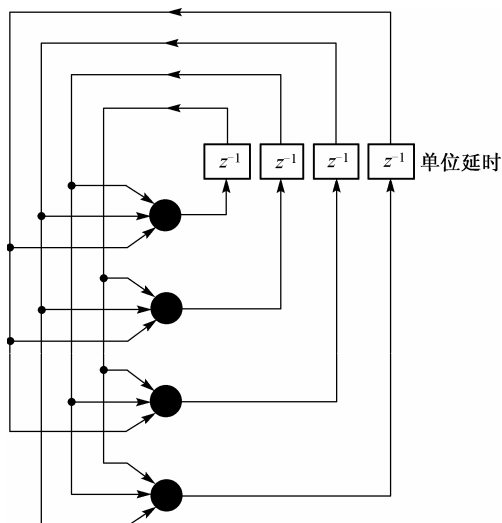


图 7.34 有 4 个单元的霍普菲尔德网络

$$\mathbf{S} = [s_1, s_2, \dots, s_N]^T$$

一般情况下，权值对称 ($\omega_{ji} = \omega_{ij}$)，且无自反馈 ($\omega_{jj} = 0$)，此时网络有以下两种工作方式：

串行（异步）方式：任一时刻只有一个单元按式(7.53)改变状态，其余单元不变，动作顺序可以随机选择或按某种确定顺序选择。

并行（同步）方式：某一时刻所有神经元同时改变状态，这种工作方式的网络常称为 Little 模型。

如果网络从 $t=0$ 的任一初始 $S(0)$ 开始变化时，存在某一有限时刻 t ，从此以后网络状态不再变化，则称网络是稳定的，即

$$S(t+1) = S(t)$$

显然，稳定状态应满足：

$$s_j = \text{sgn} \left[\sum_{i=1}^N \omega_{ji} s_i - \theta_j \right] \quad j = 1, 2, \dots, N \quad (7.54)$$

4. 玻耳兹曼 (Boltzmann) 模型 (BM 网络)

玻耳兹曼模型的概率分布采用了统计物理学中的玻耳兹曼分布，因此也称为 BM 网络。与 BP 网络相似，BM 网络具有输入层、输出层和隐含层。不同点是 BM 网络层内神经元彼此联系，其一般结构如图7.35所示。

设网络由 n 个神经元组成，每个神经元具有两个状态，并表示成

$$x_i = 0 \text{ 或 } 1, \quad i = 1, 2, \dots, n \quad (7.55)$$

设神经元之间的结合强度是对称的，即权值

$$\omega_{ij} = \omega_{ji}, \quad i = 1, 2, \dots, n \quad j = 1, 2, \dots, m \quad (7.56)$$

在网络中，神经元的输入总和发生变化将引起神经元状态的更新，这种更新在各个单元之间是非同步的，可用概率分布方法来表示。类似于热力学系统中粒子集团的变化规律，第 i 个神经元在状态更新时，新状态为 1 的概率是

$$P(x_i = 1) = \frac{1}{1 + e^{-\Delta E_i / T}} \quad (7.57)$$

式中， T 称为网络的“温度”，取正数； ΔE_i 表示第 i 个神经元的状态为 1 和状态为 0 时系统的能量差，计算公式如下：

$$\Delta E_i = \sum_j \omega_{ij} \cdot x_j - \theta_i$$

概率 P 的变化曲线如图7.36所示。一般情况下，当输入增大时，状态为 1 的概率将提高；

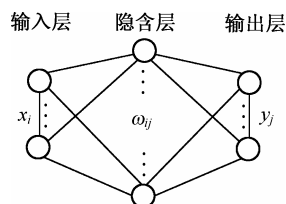


图 7.35 BM 网络一般结构

当温度 T 高时，曲线变化比较平缓；当 T 趋近于 0 时，概率 P 的曲线接近于阶跃函数。

当粒子集团接上某一热源时，最终将达到某一热平衡状态，这时集团全体的状态概率将符合玻耳兹曼分布：

$$\frac{P_{\alpha}}{P_{\beta}} = e^{-(E_{\alpha} - E_{\beta})/T} \tag{7.58}$$

式中， P_{α} 表示网络具有 α 状态的概率， E_{α} 表示该状态的能量。

由式(7.58)可以看出，当温度 $T=1$ 时，网络两个状态的出现概率的对数差等于两个状态的能量之差。

如果把求全局最优解看成是寻找最低能量状态的过程，就可以把源于物理学中退火过程的模拟退火算法用于 BM 网络的分析。

传统的启发式搜索算法，如梯度下降法等，算法每次都是向改进解的方向搜索，这种“勇往直前”的算法往往陷于一个局部最优解，而不是全局最优解，如图 7.37(a)所示。模拟退火算法是近年来新出现的一种随机搜索算法，通用性强且解的质量高，其基本思想是，在每步搜索中不但可以往改进解的方向走，而且还允许以一定的概率往恶化的方向走，即不但能接收好的状态，而且还以玻耳兹曼概率分布接收差的状态。这种“以退为进”算法的好处是在一定条件下可以收敛到全局最优解，如图 7.37(b)所示。

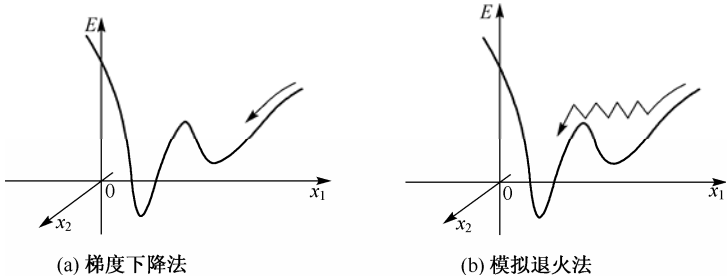


图 7.37 梯度下降法和模拟退火法搜索最优解的过程

BM 神经网络的缺点是，虽然可以在一定条件下获得全局最优解，但缓慢而细致的退火搜索过程耗费大量时间；为了求得最佳的网络权系数，需要非常多的学习周期；为了保证统计计算结果的正确，每个周期中需要很多次运算；因而，实用性受到限制。

5. CMAC 网络

CMAC (Cerebellar Model Articulation Controller) 网络是模拟小脑功能的神经网络模型，可看做一种具有模糊联想记忆特性监督式（有导师）前馈神经网络，其基本模块如图 7.38 所示。

CMAC 由一系列映射组成：

$$S \xrightarrow{e} M \xrightarrow{f} A \xrightarrow{g} u$$

式中， $S = \{\text{输入矢量}\}$ ； $M = \{\text{中间变量}\}$ ； $A = \{\text{联想单元矢量}\}$ ； $u = \text{CMAC 的输出} \equiv h(S)$ ；

$$h \equiv g \circ f \circ e.$$

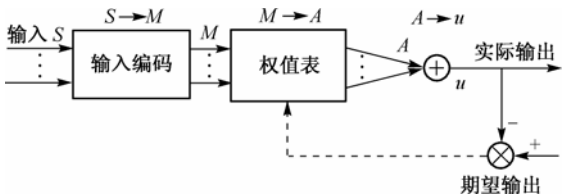


图 7.38 CMAC 网络的基本模块

(1) 输入编码 ($S \rightarrow M$ 映射)

$S \rightarrow M$ 映射是多个输入变量 s_i ($i = 1, 2, \dots, n$) 对应子映射的集合:

$$S \rightarrow M = \begin{bmatrix} s_1 \rightarrow m_1 \\ s_2 \rightarrow m_2 \\ \vdots \\ s_n \rightarrow m_n \end{bmatrix}$$

s_i 的域可通过量化函数 q_1, q_2, \dots, q_k 被近似地离散化, 每个量化函数把域分为若干间隔。对于每个 s_i 值, 都存在一个元素集合 m_i 为量化函数 q_1, q_2, \dots, q_k 的交积, 使得 s_i 值和 m_i 集合唯一地对应。

图 7.39 中, 两个输入变量 s_1 和 s_2 均由 0 至 8 域内的单位分辨率表示, s_1 的域由函数 q_1, q_2 和 q_3 描述。 q_1 把域分为 A, B, C, D 共 4 个间隔; q_2 给出 E, F, G, H 共 4 个间隔; q_3 给出间隔 I, J, K, L 共 4 个间隔。例如, $s_1=5$ 与集合 $m_1=\{B, G, K\}$ 互相映射。类似地, 值 $s_2=4$ 与 $m_2=\{b, g, j\}$ 互相映射。

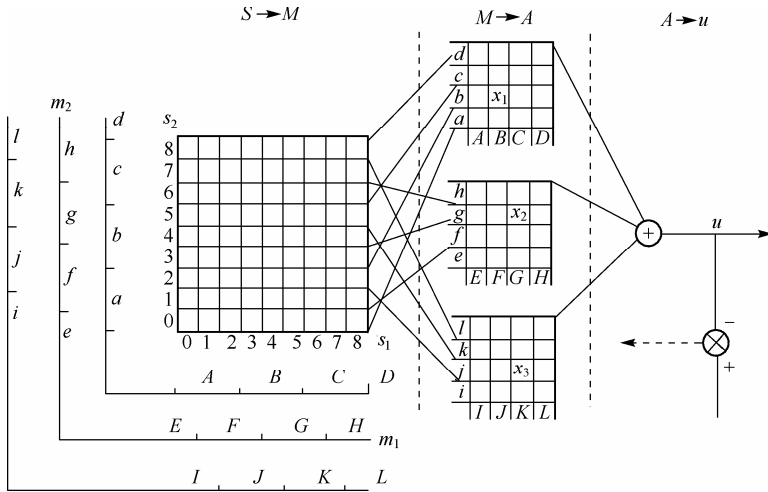


图 7.39 CMAC 网络模块内的内映射

CMAC 网络采用 $S \rightarrow M$ 映射的优点是, 可把单个精确变量 s_i 通过几个非精确信息通道传送, 每个通道只传递 s_i 的一小部分信息, 提高了信息传送的可靠性; s_i 值的较小变化不会对 m_i 内的大多数元素产生影响, 使得输入特性一般化。这对存在噪声的环境尤为重要。

(2) 地址计算 ($M \rightarrow A$ 映射)

A 为与权值表相连系的地址矢量集合, 且由 m_i 的元素组合而成。

例如, 图中集合 $m_1=\{B, G, K\}$ 和 $m_2=\{b, g, j\}$ 被组合并得到元素集合 $A=\{a_1, a_2, a_3\}=\{Bb, Gg, Kj\}$ 。

(3) 输出映射 ($A \rightarrow u$ 映射)

这一映射只对那些与地址表 A 内的地址 a_i 有关系的权值求和, 以求取网络的输出值:

$$u = \sum_i \omega_i(a_i)$$

例如，图中这些权值为

$$\begin{aligned}\omega(Bb) &= x_1, \quad \omega(Gg) = x_2, \quad \omega(Kj) = x_3 \\ u &= x_1 + x_2 + x_3\end{aligned}$$

7.3.2 人工神经网络学习规则

神经网络中的神经元必须有机地结合在一起，按照某种规律改变状态，才能体现出“智能”特征，这就要求神经元之间的连接方式和权系数必须按一定的规律安排。又由于神经网络一般都很大、很复杂，它的连接方式和权系数不可能事先人为地设置好，因此，神经网络必须能自学习或自组织，形成所需要特性的结构。

神经网络的学习可以定义为学习是状态的变化，以达到性能上有意义的稳定性，这种状态变化是由反复作用、训练或实验而得到的。学习规则是指神经网络学习的一套精确算法，用以确定神经元应如何改变它的连接权重，从而正确响应训练。目前已有许多种学习规则，例如，常用的 Hebb 规则、 δ 规则、ART 规则和玻耳兹曼规则等。

1. 赫布 (Hebb) 规则

赫布规则是较早提出的一种神经网络学习算法。在此规则的基础上，出现了多种形式的学习算法。

赫布规则可归纳为一切联想（长期的）记忆都保存在中枢神经系统的突触连接中，而且所有学习均意味着突触耦合强度的变化。当某一突触（连接）两端的神经元同步激活（同为激活或同为抑制）时，该连接的强度应增强，反之应减弱。它是相联式学习的基本心理学定律，也是人工神经网络的基本定律。通常可用数学方式描述为

$$\begin{aligned}\Delta\omega_{i,j} &= \eta \cdot a_i \cdot a_j \\ \omega_{i,j} &= \eta \cdot \sum_{u=0}^{p-1} a_i^u \cdot a_j^u\end{aligned}\tag{7.59}$$

式中， $\omega_{i,j}$ 为第 j 个神经元到第 i 个神经元的连接强度（权系数）； $\Delta\omega_{i,j}$ 表示权系数的变化； a_i, a_j 为神经元 i 和 j 在输入样本 P （ $u = 1, 2, \dots, p$ ）时的活化状态； η 为学习速度系数。

赫布学习规则的优点表现在神经元 i 和 j 的激励值 a_i, a_j 的相关性，即能产生有用的关联学习。对于某一特定的神经元 i ，当其处于活化状态时，它将倾向于激励那些曾与其相关的活化神经元。由于 $\Delta\omega_{i,j}$ 与 a_i, a_j 的相关成比例，有时也称为相关学习规则。

2. δ 规则

δ 规则是赫布规则的延伸，是目前最为流行的神经网络学习算法。 δ 规则学习的最终目的是通过权系数的调整，纠正实际输出，使基于误差信号的某一目标函数达到最小，以使网络中每一输出单元的实际输出在某种统计意义上逼近应有输出。

设输入为 x_k 时，神经元 k 的实际输出 y_k 与希望输出 d_k 之间的误差为

$$e_k = d_k - y_k\tag{7.60}$$

对于宽平稳的学习过程，最常用的目标函数是均方误差判据：

$$J = E \left[\frac{1}{2} \sum_k e_k^2 \right]\tag{7.61}$$

式中， E 为求期望算子。

直接用 J 作为目标函数时，需要知道整个过程的统计特性，通常用 J 在时刻 n 的瞬时值 $\xi(n)$ 代替 J ，即

$$\xi(n) = \frac{1}{2} \sum_k e_k^2 \quad (7.62)$$

求 $\xi(n)$ 对权值 ω 的极小值，根据梯度下降法：

$$\Delta \omega_{kj} = \eta \cdot e_k \cdot x_j \quad (7.63)$$

式中， η 为学习步长。

δ 规则属于有教师学习（监督学习）规则，需要外界的“教师”对给定一组输入提供应有的输出结果（正确答案），这组已知的输入-输出数据称为训练样本集。

3. ART 规则

自适应共振理论（Adaptive Resonance Theory, ART）是一种无教师学习（非监督学习）规则，系统完全按照环境提供数据的某些统计规律来调节自身参数或结构（这是一种自组织过程），以表示出外部输入的某种固有特性（如聚类或某种统计上的分布特征）。

ART 模型中一般有三层，其中输入层和输出层称为短期记忆层，用以把握当前的输入模式、模式的分类和所期望的模式；中间层称为长期记忆层，已学模式以权重方式存储在长期记忆层中；输出层到输入层存在一个反馈矢量，称为由顶向下的自稳机制，以保证长期记忆层对已学到东西的记忆具有足够的牢固性。每当一个模式输入到系统，所期望的模式便从长期记忆中调出所期望的模式加以比较，如果输入的模式与所期望的各模式之一相类似，网络就会把输入模式归入对应类中。如果没有相似的，则形成一个新的输入矢量表示类别，并作为首次记忆保存。

ART 的非指导学习方法又称为“竞争学习”：让神经元为一个有限量的活性值而相互竞争，最简单的竞争学习是所谓“胜者全取”，即只有一个最大输入的神经元被激活，其余的神经元被抑制，可表示为

$$\Delta \omega_{ji} = \begin{cases} \eta(x_i - \omega_{ji}) & \text{若神经元 } j \text{ 竞争获胜} \\ 0 & \text{若神经元 } j \text{ 竞争失败} \end{cases} \quad (7.64)$$

ART 使用分类法来选择输入模式的类。当给出一个输入模式选中一个优胜者时，网络就通过反复调整长期记忆的权重的办法来处理输入模式，并将输入模式的特征加到长期记忆之中。按照上面给出的规则，只有当新的输入模式与存入记忆中的某个老的模式足够相似时，二者才能融洽，即有关的权重系数进行调整，从而使长期记忆得以改变，造成一种自适应谐振状态，这也就是其名称的由来。

ART 规则是一个比较接近人脑处理信息方式的学习规则，其优点是：

- 可以“实时学习”，即学习和工作是不分开的；
- 自治和自组织的学习，能够达到稳定、可靠的结果；
- 可以有选择地学习，即形成“集中注意”的状态，因而具有较高的学习效率；
- 可以完全避免陷入局部最小点问题。

其不足之处在于过分敏感，小的输入变化就可能使输出变化很大。

4. 玻耳兹曼规则

玻耳兹曼规则是一种加强式学习规则，用一个目标函数来估价网络的输出。

加强式学习规则介于教师学习、无教师学习规则之间，外部环境对系统输出结果给出评价信息（奖或惩），而不是给出正确答案。学习系统通过强化那些受奖的动作来改善自身性能。

神经网络的学习就是改变网络中各个权重系数的数值。玻耳兹曼学习规则独特之处在于它用了个概率函数来改变连接权重。正因为权重改变式是一个概率函数，故又称这种网络是“随机网络”。

设系统的能量函数为

$$E_i = -\frac{1}{2} \sum_{i,j} \omega_{ij} \cdot x_i \cdot x_j \tag{7.65}$$

改变权重系数 ω_{ij} 会导致 E 的变化。由前式(7.58)可知， E 的变化直接引起系统中各种状态（尤其是其中的各个能量极小点）出现概率的变化。

系统可以在三种情况下运行：

- 学习情况：系统与外界接触的神经元全部受到约束；
- 工作情况：与外界接触的神经元中有部分受到约束；
- 自由状态：所有神经元都不受约束。

在玻耳兹曼学习规则中，神经网络中神经元状态被视为类似某种物质体的微观状态，而温度 T 是改变网络权重的参数。在学习情况下调整各个系数，使得对于各种特定的约束而言，在学习和工作两种情况下，系统通过搜索能够达到一致或非常接近的热平衡状态。事实上，这些状态就是系统能量函数中的一些极小点，通过学习后，它们都具有某种特定的出现概率。在自由状态下，系统通过搜索达到上述状态也应具有一定的概率。

这种学习规则一般包括以下几个步骤：

- ① 将所有的网络权重随机化，并设初始温度为高温 T ；
- ② 用输入矢量和网络输出值计算目标函数的值，即计算输出值是否符合目标；
- ③ 根据玻耳兹曼分布 $P(x) = e^{-x^2/T^2}$ ，随机地改变每一权重；
- ④ 重新计算目标函数；
- ⑤ 如果目标函数已得到改善，就将上述权重改变确定下来；如果目标函数未得到改善，则降低温度 $T(n+1) = T(0)(1/\log(1+n))$ ，回到步骤②。

温度在网络初始化时高，权重改变概率大。随着网络被训练而降低温度，使权重每一次改变量越来越小，网络逐步趋向最佳状态。目标函数引导网络向最佳状态逼近，正确的方向得到加强。

7.3.3 人工神经网络控制的结构方案

1. NN 监督式控制

如图7.40所示，NN 监督式控制的结构中包括一个监督程序和一个可训练的神经网络控制器（NNC）。控制器的输入对应于由人接收（收集）的传感输入信息，而用于训练的输

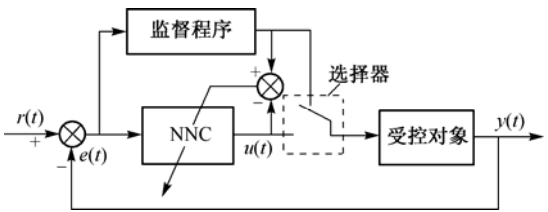


图 7.40 基于神经网络的监督式控制

实现 NN 监督式控制的步骤如下：

- ① 通过传感器和传感信息处理，调用必要的和有用的控制信息；

- ② 构造神经网络，选择 NN 类型、结构参数和学习算法等；
 - ③ 训练 NN 控制器，实现输入和输出间的映射，以便进行正确的控制。在训练过程中，可采用线性律、反馈线性化或解耦变换的非线性反馈作为导师（监督程序）来进行 NNC 的训练。
- 在标准的倒立摆控制系统中采用了 NN 监督式控制。

2. NN 自适应判断控制

NN 自适应判断控制应用了加强式学习规则，通常由自适应判断网络 AJN 和控制选择网络 CSN 组成，如图7.41 所示。

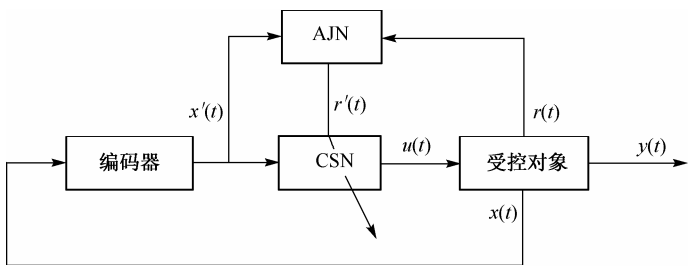


图 7.41 NN 自适应判断控制

- AJN 相当于强化学习需要的“教师”，它的作用是
- 通过不断的奖罚强化学习，使 AJN 逐渐训练为一个熟练的教师；
 - 经过学习后的 AJN，根据反映受控系统当前状态的编码信号 $x'(t)$ 和外部反馈信号 $r(t)$ ，判断当前控制作用的效果；然后向 CSN 提供内部强化信号 $r'(t)$ 。
- CSN 在内部强化信号的引导下进行学习，相当于多层前馈神经网络控制器。通过学习，CSN 根据系统编码后的状态，选择下一个控制作用。

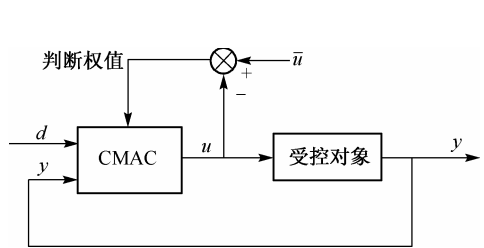


图 7.42 控制与学习分阶段进行的 CMAC 控制

3. 基于 CMAC 的控制

- (1) 控制与学习分阶段进行的 CMAC 控制
- 如图7.42 所示，作为闭环控制系统，除了指令变量外，反馈变量也加以编码，用做 CMAC 模块的输入。控制器的输出直接送至受控对象，控制器的训练基础是期望输出和实际输出间的差别，因而必须提供控制器的期望输出。

系统工作分如下两阶段进行：

- 训练阶段：当 CMAC 接收到指令和反馈信号后产生一个输出，此输出与期望输出进行比较，如果两者存在差别，则调整权值以消除差别。训练可看做对一个适当的反馈控制器的辨识。经过这一阶段，CMAC 已经学会如何根据给定指令和所测反馈信号产生合适的输出，用于控制受控对象。
 - 控制阶段：当需要的控制接近所训练的控制要求时，CMAC 就能够很好地工作。
- 这两个阶段都无须分析装置的动力学和求解复杂的方程式，但权值判断是以控制器期望输出与控制器实际输出间的误差（而不是装置的期望输出与装置的实际输出间的误差）为基础的。因此在训练阶段，要求期望的装置输入即控制器的期望输出是已知的。

(2) 控制与学习同步进行的 CMAC 控制

如图 7.43 所示，需要为受控装置设计一个固定增益常规偏差反馈控制器，参考输出模块

在每个控制周期产生的期望输出被送至 CMAC 模块, 作为对固定增益控制器控制信号的补充。在每个控制周期之末, 执行一步训练。在前一个控制周期观测到的装置输出用做 CMAC 模块的输入。用计算输入 u^* 与实际输入 u 之间的差来计算权值。当 CMAC 跟随连续控制周期不断训练时, CMAC 函数在特定的输入空间域内形成一个近似的装置逆传递函数。如果未来的期望输出在域内相似于前面预测的输出, 则 CMAC 的输出也会与所需的装置实际输入相似, 输出误差将很小。经训练后, CMAC 成为主控制器, 接替固定增益常规控制器。

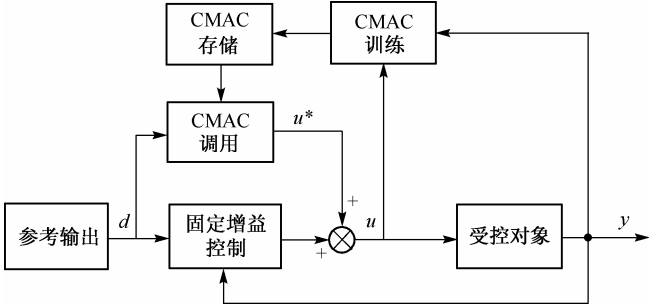


图 7.43 控制与学习同步进行的 CMAC 控制

如果未来的期望输出在域内相似于前面预测的输出, 则 CMAC 的输出也会与所需的装置实际输入相似, 输出误差将很小。经训练后, CMAC 成为主控制器, 接替固定增益常规控制器。

4. 多层神经网络控制

多层神经网络控制器基本上是一种前馈控制器。一个普通的多层神经控制系统如图 7.44 所示。该系统存在两个控制作用: 前馈控制和常规反馈控制。前馈控制由神经网络实现, 训练目标是使期望输出与实际装置输出间的偏差最小, 该误差作为反馈控制器的输入。反馈作用与前馈作用被分别考虑。

多层 NN 控制器有 3 种结构: 间接结构、通用结构和专用结构。

(1) 间接学习结构

图 7.45 所示的间接多层 NN 控制结构含有两个同样的神经网络, 用于训练。每个网络作为一个逆动态辨识器。训练的目标是要根据期望响应 d 找到一个合适的控制 u 。以网络 I 和网络 II 间的差为基础来调整权值, 使得误差 e 为最小, 则 $u = u^*$, 但这并不保证期望输出 d 与实际输出 y 之间的差别为最小。

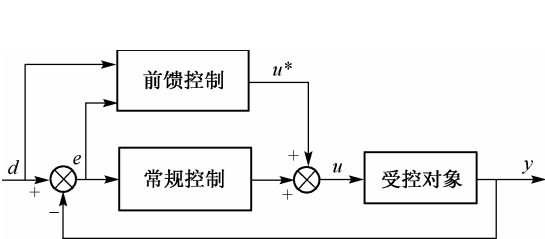


图 7.44 多层 NN 控制的一般结构

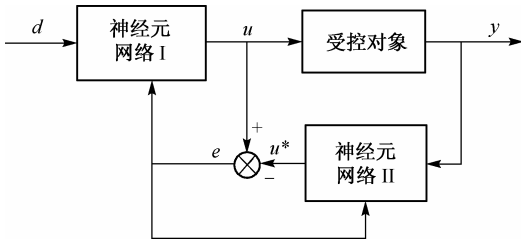


图 7.45 间接学习结构

(2) 通用学习结构

图 7.46 给出的多层 NN 控制的通用学习结构可以使期望输出 d 与实际输出 y 之间的差最小。通过训练, 使得装置输入 u 与网络输出 u^* 间的差别最小。在训练时, u 应当处在使 y 覆盖期望输出 d 的范围内。训练之后, 如果某一期望输出 d 被送至网络, 则该网络就能够为受控装置提供一个合适的 u 。

本结构的局限性是, 一般无法知道哪一个 u 对应于期望输出 d , 因而网络不得不在 u 的大范围内进行训练, 以求经过学习能够使装置输出 y 包括期望值 d 。

(3) 专用学习结构

多层 NN 控制的专用学习结构如图 7.47 所示。当神经网络训练时, 期望输出 d 是该网络

的输入。采用误差反向传播方法，经过训练使期望输出 d 与装置的实际输出 y 之间的差别 e 最小。因此，不仅能够期望得到良好的装置输出，而且训练能够在期望输出范围内执行，而不需要知道装置的合适输入范围。不过，本结构中把装置作为网络的一层来处理。为了训练该网络，必须知道装置的动力学模型，或者进行某种近似处理。对多层神经网络控制器的训练是由误差反向传播训练算法来完成的。该误差可以是期望输出与实际装置输出间的差，也可以是校正装置输入与由神经网络计算得到的输入之间的差。

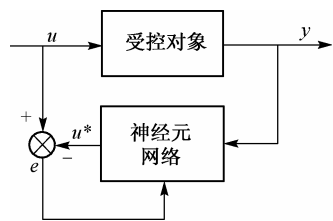


图 7.46 通用学习结构

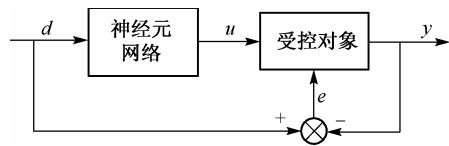


图 7.47 专用学习结构

5. 分级神经网络控制

(1) 分级控制模型的组成

如图7.48所示，基于神经网络的分级控制模型由 3 部分组成。

① 常规控制器，它是以期望装置输出 d 与由传感器测量的实际装置输出 y 间的误差 $e=(d-y)$ 为基础的反馈控制，通常是比例微分控制器。

② 与神经网络 I 连接的通道。网络 I 是学习受控对象特性的动力学模型，当接收到装置的输入 u 时，经过训练的神经网络 I 输出 y^* 能够近似装置的输出 y ，起到系统动态特性辨识器的作用。该通道以误差 $d-y^*$ 为基础，因为没有传感滞后，作用比常规外反馈回路快得多。

③ 神经网络 II。神经网络 II 监控期望输出 d 和装置输入 u ，学习建立装置的逆动力学模型。当它收到期望输出指令 d 时，经过训练，产生一个合适的装置输入分量 u^* 。

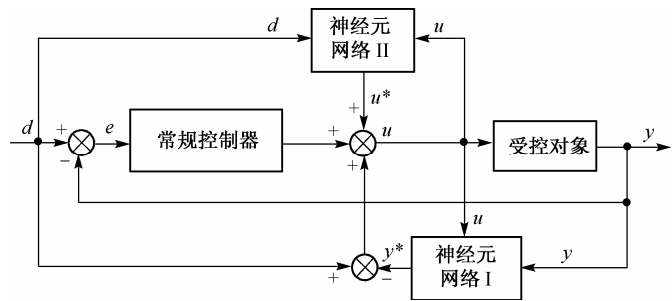


图 7.48 分级神经网络控制器

(2) 分级神经网络模型的运作过程

控制与学习同步执行。两个神经网络起到辨识器的作用，其中一个用于辨识装置动力学特性，另一个用于辨识逆动力学特性。

传感反馈主要在学习阶段起作用，此回路提供一个常规反馈信号去控制装置。由于传感延时作用和可允许控制增益较小，系统的响应较慢，从而限制了学习阶段的速度。

在学习阶段，神经网络 I 学习系统动力学特性，而神经网络 II 学习逆动力学特性。随着学习的进行，内反馈逐渐接替外反馈的作用，成为主控制器。

然后，当学习进一步进行时，该逆动力学部分将取代内反馈控制，最后结果是该装置主要由前馈控制器进行控制，因为装置的输出误差与内反馈都几乎不复存在，从而提供处理随机扰动的快速控制。

分级神经网络模型控制系统可以分为两个系统，即基于正向动力学辨识器的系统（如图7.49所示）和基于逆向动力学辨识器的系统（如图7.50所示），也可以单独应用它们。

- (3) 分级神经网络模型的控制系统的优点
- ① 该系统含有两个辨识器，一个用于辨识受控对象的动力学特性，另一个用于辨识受控对象的逆动力学特性；
 - ② 有一个主反馈回路，对训练神经网络十分重要；
 - ③ 当训练进行时，逆动力学部分逐步变为主控制器；
 - ④ 控制的最后效果与前馈控制的效果相似。

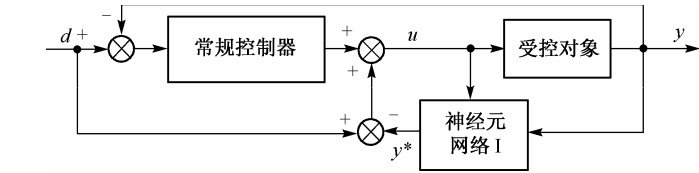


图 7.49 基于正向动力学辨识器的控制系统

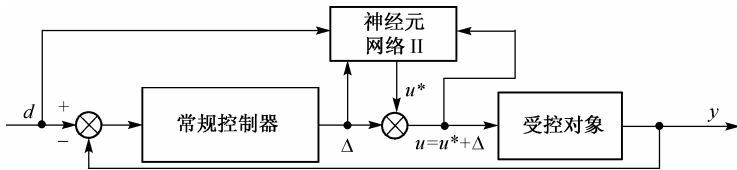


图 7.50 基于逆向动力学辨识器的控制系统

7.3.4 人工神经网络控制的特点

从数学本质来看，人工神经网络是有足够大量冗余的待定系数的多输入输出方程组，通过多次迭代确定系数达到收敛，因而，人工神经网络具有下列适用于控制的特性。

- 1. 良好的非线性系统的控制功能**
具有表达非线性系统的能力和自适应学习功能，对非线性系统的控制性能好。
- 2. 学习和自适应性**
人工神经网络可以从案例中学习，训练好的网络在训练的范围内对训练（学习）时没有出现的输入信号有较好的预报能力，网络也能在线自适应地学习。
- 3. 并行分布处理**
由于网络的并行分布结构，处理速度较快，抗故障能力较高。
- 4. 多变量系统**
网络本身具有多输入/多输出的结构，因此比较容易适用于多变量系统。
- 5. 数据种类多样化**
网络能同时处理定量和定性的数据，因此在系统辨识/预报中有广泛的应用。
目前，人工神经网络控制在实际应用中最大的不足在于实时性和收敛性问题。

7.4 多种控制策略的渗透和结合

为了更好地满足不同应用的不同要求，各种控制策略取长补短，互相渗透，结合成复合控制模式是必然的发展趋势。

模糊控制、专家系统、神经控制都可与其他传统控制和现代控制组合起来，形成新的控制集成，例如：

- 与 PID 控制复合形成的模糊 PID (FI)、专家 PID (EI)、模糊专家 PID (FEI)、神经网络 PID (NI)；
- 与史密斯控制复合形成的史密斯模糊控制 (SF)；
- 与解耦控制复合形成的多变量解耦模糊控制 (DF)；
- 与自适应控制复合形成的自适应模糊控制 (AF)、自适应神经网络控制 (AN)；
- 与变结构控制复合形成的模糊变结构控制 (FV)、神经网络变结构控制 (NV)；
- 与预测控制复合形成的模糊预测控制 (FP)、专家预测控制 (EP)，等等。

智能控制的多种控制策略互相结合成的复合控制策略，也获得了广泛的应用。

7.4.1 模糊神经网络 (FNN) 控制

模糊逻辑与神经网络是实现语言知识表示和自适应知识发展这两种人类控制基本特征的较好工具，由于模糊控制本质上是变结构和非线性控制，而神经网络通过训练和学习能自动或半自动提取隐含的控制规则，并用数值计算方法来描述复杂非线性控制的行为。因此，二者特性的互补克服了模糊控制中人工获取复杂对象控制规则的困难，使一个系统能够更像人类一样处理认知的不确定性。

模糊神经网络有不同的类型，在正则模糊神经网络 (RFNN) 中，输入信号或权值是模糊的，或者输入信号和权值都是模糊的；在混合模糊神经网络 (HFNN) 中，组合模糊信号和神经网络权值，应用加、乘等操作获得神经网络输入。

需要外部教师信号的监督式（有导师）学习，以及只靠神经网络内部信号的非监督式（无导师）学习算法，可直接推广至 FNN。

目前，模糊神经网络控制主要可分为如下两类系统：

- 分别应用相应的神经网络使它们在结构上等价于模糊器、模糊规则库和解模糊器，直接实现模糊控制系统。但这种方法中，原来的模糊系统的知识结构受到学习速度缓慢的约束；
- 通过神经网络，在功能上等价于模糊系统。

第二种系统的集成方法如下。

1. 控制器结构

由于模糊系统和神经网络在非线性映射和近似能力方面的相似性，可以把神经网络的学习能力集成到模糊系统中。采用多层前向神经网络来实现模糊系统内的映射，具有两个输入变量和一个输出变量的模糊神经控制器的结构如图7.51所示。

图7.51中，模糊神经系统第1层的节点为输入节点（语言节点），表示运行过程的输入语言变量，而且与第2层没有权值关系；

第2层由隶属函数节点组成，表示输入语言变量的全部模糊集，并完成从精确输入值至模糊值的映射；

第3层为中间层，其节点没有明确的意义；

第4层的节点表示输出变量 $(-6 \sim +6)$ 离散域内的点。连接权 $[\omega_{ij}]$ 和 $[\omega_{jk}]$ 被训练来表示控制规则。在第3层和第4层的节点的实际函数为S函数 $f(x) = 1/(1+e^{-x})$ ；

第5层为输出层，执行解模糊过程。

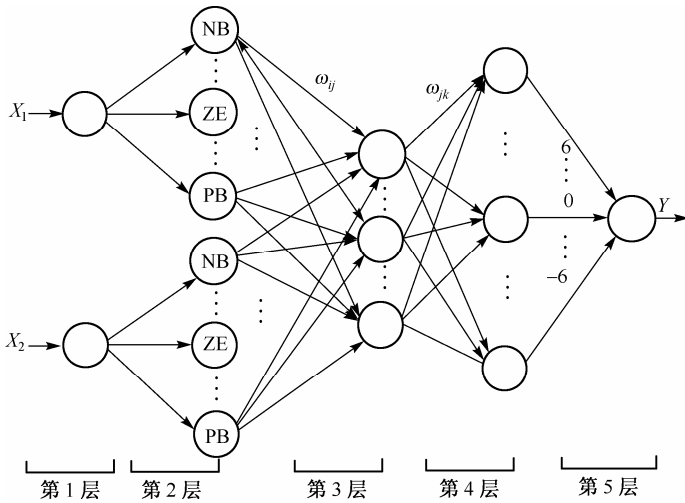


图 7.51 两输入和一输出的模糊神经控制器结构

在这个5层结构的连接机制模型中，模糊系统从模糊化、模糊推理到解模糊化的全过程，都能够通过神经网络的正向计算来执行。

2. 建立模糊关系

模糊系统的模糊关系，即模糊规则库能够通过学习程序用神经网络的权值并行存储。假定模糊控制器具有两个输入 X_1 和 X_2 和一个输出 Y 。它们对应的模糊集 A, B 和 C 定义为 $\{NB, NM, NS, ZE, PS, PM, PB\}$ ， A, B 的隶属函数为单点型， C 的隶属函数为三角形，如图7.52所示。

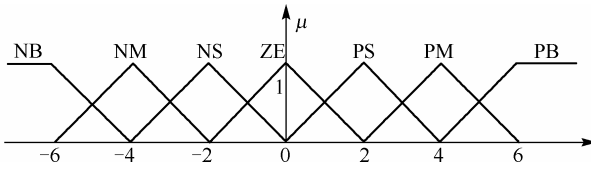


图 7.52 模糊神经系统的隶属函数

模糊规则库包含6条规则：

- R_1 : if X_1 is PB and X_2 is PB then Y is PB
- R_2 : if X_1 is PM and X_2 is PB then Y is PM
- R_3 : if X_1 is PS and X_2 is PS then Y is ZE
- R_4 : if X_1 is PM and X_2 is NB then Y is NM
- R_5 : if X_1 is PS and X_2 is NM then Y is NS
- R_6 : if X_1 is PS and X_2 is NS then Y is ZE

与输入模糊集的隶属函数相对应的第2层输入可表示为

$$[\mu_{NB}(a), \mu_{NM}(a), \cdots, \mu_{PB}(a), \mu_{NB}(b), \cdots, \mu_{PM}(b), \mu_{PB}(b)]$$

第 4 层的输出为输出模糊集的隶属函数，可表示为

$$[\mu_C(-6), \mu_C(-5), \cdots, \mu_C(-1), \mu_C(0), \mu_C(1), \cdots, \mu_C(5), \mu_C(6)]$$

各条规则可以用对应的训练样本表示。例如，规则 R_2 的训练样本为

输入样本: [0, 0, 0, 0, 0, 1, 0; 0, 0, 0, 0, 0, 0, 1]

输出样本: [0, 0, 0, 0, 0, 0, 0, 0, 0.5, 1, 0.5, 0]

采用误差反向传播学习算法，基于上述学习样本来训练神经网络，可以将所有模糊规则作为网络的权来保持。规则的增加和更新可由增加和更新训练数据集来完成，计算负担也较轻。

3. 模糊推理

模糊推理由基于下列两种原理的模糊神经网络的并行计算来实现：

(1) 当输入模糊集 A 和 B 与样本 A_k 和 B_k 相似时，模糊蕴含 $A_k, B_k \rightarrow C_k$ 被激发，输出模糊集 C 与 C_k 相似。

(2) 当输入模糊集与样本模糊集不同时，一个模糊蕴含序列将被激发至不同程度，输出是对应的被激发规则的非线性插值。

4. 模糊神经网络与其他控制策略的结合

集神经网络和模糊控制之所长的一个成功实例是，采用两个 BP 神经网络实现误差及其变化率的隶属函数生成，第三个神经网络产生模糊规则输出，在数字信号处理器（DSP）及微机上实现，构造一个非线性自整定控制器，不仅设计简单，系统无超调和鲁棒性强，而且控制性能优于传统的 PI 控制。

在模糊神经网络的基础上，再与其他控制策略结合，可以形成更高层次的性能更优良的控制策略。例如：

(1) 模糊神经网络变结构控制（FNV）

发挥三者之长，可形成比模糊控制和变结构控制动静品质更佳，设计更简单，而且比模糊神经网络学习收敛速度更快的控制策略。

(2) 模糊神经自适应控制（FNA）

在自适应控制中，引入模糊神经网络建模工具，可以改善自适应控制的鲁棒性和实时性，特别适用于具有不确定性的非线性系统跟踪控制，也可用于实际的伺服直流电机调速控制。

7.4.2 神经网络专家系统

1. 专家系统的不足

专家系统将问题限于某一特定的专门领域，降低了问题的无限复杂性，减少了系统处理与理解能力的有限性之间的矛盾。但仍然存在不少问题，主要有

(1) 知识的获取

专家系统本质上是一个知识工程系统。知识获取、知识表示、知识存储和知识运用构成了专家系统工作的主要内容，而知识获取是普遍存在的难题。专家系统的知识来源于领域专家，如果由知识工程师向领域专家获取，则经常遇到的问题是领域专家自己也很难把自己的知识、经验或直觉讲清楚，甚至有时专家“知识”可能有误；如果由机器自动向“专家”获取，则受到目前机器学习能力仍相当低下的限制。

(2) 知识的“高台效应”

专家系统的专业领域相对比较狭窄，如果超出专家知识的适用范围，则系统就不能正确

处理；而专家系统本身往往并不能判断什么时候或什么情况下问题已接近或超出了它的边界。

(3) 知识存储容量与运行速度的矛盾

为了更好地处理领域问题，总希望有大量的知识存储和大规模的搜索、匹配，然而这样解决问题就很慢，从根本上限制了专家系统解决大型复杂问题的能力。

神经网络专家系统可以利用神经网络系统的自学习功能、联想记忆功能、分布式并行信息处理功能来解决上述问题。

2. 神经网络专家系统的原理

神经网络专家系统的基本原理是，利用神经网络并行推理的特性使专家系统具有联想记忆的能力，不需要建立从输入模式 x 到输出结论 y 之间的推理规则，而是从例子（或训练集）中学习并修正，因而不仅记住了已给实例，而且抓住了可以从特定事例推广到更一般性原理（规则）的特征，用来区分已知实例，还可以用来推理、分析未知实例。

神经网络专家系统利用神经网络的分布性，突破了专家系统中知识获取这个瓶颈问题。与传统计算机局域式信息处理方式不同，神经网络在进行知识获取时，只要求专家提供范例及相应的解，通过特定的学习算法对样本进行学习，经过网络内部自适应算法，不断修改权值分布以达到要求。把专家求解实际问题的启发式知识和经验分布到网络的互连及权值上，建立起特有的推理模型，从而完成知识的表达。

对于特定的输入模式，神经网络通过前向计算，产生一个输出模式，其中各个输出节点代表的逻辑概念同时被计算出来；通过输出节点和本身信号的比较得到特定解，同时排除其他解。在神经网络中，允许输入失真的学习样本，只要输入接近某一学习样本的输入模式，输出就会接近学习样本的输出模式。

神经网络专家系统利用神经网络的学习功能、大规模并行分布处理功能、连续时间非线性动力学和全局群体作用，实现知识获取自动化，克服“组合爆炸”、“推理复杂性”及“无穷递归”等困难，实现并行联想和自适应推理，达到更高的智能水平及实时处理能力。

3. 神经网络专家系统的基本结构

神经网络专家系统的基本结构如图 7.53 所示，其中，自动知识获取模块获取专家知识；推理机制提出使用知识去解决问题的方法；解释模块说明专家系统是根据什么推理思路作出决策的；I/O 系统是用户界面，提出问题并获得结果。

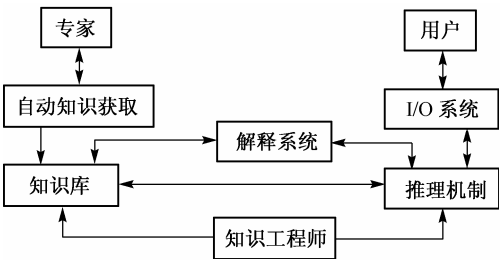


图 7.53 神经网络专家系统的基本结构

(1) 知识获取

包括提出所需神经网络的结构（包含网络层数、输入、输出及隐含节点个数），组织待训练的学习样本，使用神经网络学习算法，通过样本学习，得到所需权值分布，从而完成知识获取。

(2) 知识库

由自动知识获取构筑的知识库是推理机制完成推理和问题求解的基础。知识库可以不断创新，即在原基础上对新样本进行学习，获得具有更多知识与经验的新的网络参数。

(3) 推理机制

与传统的专家系统所用的基于逻辑的演绎方法不同，神经网络专家系统的推理机制是一个数值计算过程，主要由以下三个部分组成：

- ① 输入逻辑概念到输入模式的变换，并根据领域的特点，确定变换规则，再将目前的状态变换成神经网络的输入模式；
- ② 通过网络内的前向计算产生神经网络的输出模式；
- ③ 输出模式解释：将输出数值向量转换成高层逻辑概念，不同领域的解释规则各异。

4. 模糊神经网络专家控制（FNE）

将专家系统的灵活性和集成性用于模糊神经网络控制中，在初始阶段作辅助控制，间接缓解对神经网络快速学习的要求，并可通过专家系统方法直接改进神经网络学习问题，得到理想实用的工业控制器。

7.5 基于进化和群体智能的控制算法

仿生学的研究人员从生物进化机理中受到启发，提出了许多用以解决复杂优化问题的新方法，如遗传算法、进化规划、进化策略等。对生物演化和生物群落行为的研究导出了基于进化和群体智能的控制算法。作为优化技术领域的一个研究热点，研究社会性生物的自组织行为，对群体智能进行数学建模，并用计算机对其进行仿真而创立的群体智能算法，与进化策略及遗传算法有着特殊的联系。典型的社会性生物的自组织行为包括蚁群的筑巢、觅食、迁徙，鸟群的觅食、共同飞行等行为。单只蚂蚁的能力是有限的，但是当这些简单的蚂蚁组成蚁群时，却能够完成筑巢、觅食、迁徙等复杂行为。鸟群在没有集中控制的情况下能够同步飞行。大量简单的个体组成的群体合作可以完成非常复杂的任务，比由复杂个体完成的方法更具有健壮性、灵活性和经济上的优势。

7.5.1 遗传算法

1. 遗传算法的基本思想

遗传算法是基于自然界遗传和进化规律，模拟基因进化过程的搜索算法。

遗传算法的基本思想源于达尔文的进化论，将待求解问题转换成由个体组成的演化群体和对该群体进行操作的一组遗传算子，整个系统按照“物竞天择，适者生存”的原则，经历生成—评价—选择—操作的演化过程反复进行，直至搜索到最优解。

遗传算法和神经网络的相同点在于都是基于生物机制的方法，其不同在于神经网络是模拟人的大脑机制，遗传算法则是模拟生物的进化机制。

遗传算法对解的染色体编码与操作，结合已有信息和搜索空间的新领域来产生新的结果，能利用已有解编码中较好的“建筑块”自动生成更好、更完整的解编码，然后循环地利用选择、交叉和变异等遗传操作对种群集合进行更新，种群集合中可行解的优劣采用适应值来进行评估，采用概率的变迁规则来指导它的搜索方向，直到算法的终止条件满足为止。

2. 遗传算法的工作步骤

遗传算法的工作步骤如图7.54所示。

(1) 编码

把欲优化的参数

$$x \in [X_{\min}, X_{\max}]$$

用二进制编码成字符串，即有

$$x = X_{\min} + \frac{\sum_{i=1}^L g_i 2^i}{2^{L+1} - 1} (X_{\max} - X_{\min})$$

式中, L 为字串的长度; $g_i = 0$ 或 1 为第 i 个基因的值。

每个二进制代码位即为基因, 字串即为个体或称为染色体。一定数量 N 的个体形成群体。随机产生 N 个个体作为原始群体 $P(0)$ 。

(2) 评价

对每个个体, 计算适合度函数 F_i , $i = 1, \dots, N$; 以此来评价群体 $P(k)$ (k 为进化过程中的繁衍代数) 中的所有个体。适合度函数可取为目标函数或另行构造的可对染色体进行评价的函数。

(3) 复制 (选择)

按串的复制概率 $p_c = F_i / \sum_{i=1}^N F_i$ 从群体 $P(k)$ 中选取用以产生子代 $P(k+1)$ 的个体。用轮转法随机地在 $(0, 1)$ 区间中任取一点, 适合度越大的个体, 被选中的概率也越大, 从而得以生存; 反之, 适合度小的个体则被淘汰。这样, 选出的 N 个个体进入匹配池。

(4) 交叉

随机将选中的个体两两配对, 在字串上随机选取两个位置, 并按交叉概率 p_c 将此位置内的代码交换, 经交叉后得到的子代新个体, 进入繁衍池。遗传算法的有效性主要来自复制和交叉, 尤其是交叉起着核心的作用。

(5) 突变

按突变概率 p_m , 随机选取若干个体, 并随机选取其中的一个求反。突变的作用是保持群体中基因的多样性。至此, 新一代群体 $P(k+1)$ 产生, 进入群体池。

(6) 重复

重复(2)~(5), 直至群体的总适合度稳定。此时, 群体中的个体具有相同的基因, 并为最优解。

3. 遗传算法运行机制的主要特点

① 遗传算法是对问题参数的编码群 (染色体) 进行进化, 而不是对参数本身, 因此不受被优化函数约束 (例如, 连通性、凸性、连续性、导数存在等) 的限制, 也不受搜索空间的限制。

② 遗传算法是在字串群体中进行搜索, 而不是在单个点上寻优。这样, 可以大大减少陷入局部优化和局部收敛的可能性, 具有全局快速收敛的特点。

③ 遗传算法仅使用问题本身所具有的目标函数或其适应度进行工作, 而不需要任何先决条件和其他信息, 因此可以处理的问题十分广泛, 如多目标、非线性或基于知识的目标函数的优化。

④ 遗传算法使用随机规则进行操作, 而不是某个确定性的规则, 因此可以很快地到达最优解附近。

⑤ 遗传算法具有隐含的并行性, 它使用相对少的字串, 就可以在数量相当大的区域中完成搜索。已经证明, 当每一繁衍代对 N 个染色体进行操作时, 实际上处理了 $N \times N \times N$ 个模式。

4. 遗传算法的应用

遗传算法在自动控制中的应用主要是进行优化和学习, 特别是与其他一些控制策略结合, 能获得很好的结果。

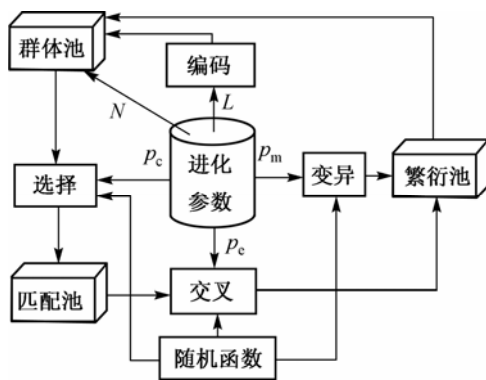


图 7.54 遗传算法的工作步骤

由于遗传算法的强健性和对复杂问题的求解能力，已经有学者将遗传算法应用于同构系统或异构系统中的任务调度，以及发展了并行遗传算法解决调度问题。遗传算法具有强大稳健的隐并行解空间搜索功能，因此可用来开发快速可伸缩的任务分配与调度策略的最优求解算法。目前已经开发了各种利用遗传算法调度的串并行算法，在解决复杂组合问题上具有优越性，能得到较好的效果，但设计的参数较多，实现比较困难，运行时间长。

同时，遗传算法在求解优化问题的许多策略上值得其他算法借鉴。

虽然由于实时性问题，目前遗传算法的实际工业应用还不多，但其深刻的思想和优异的性能有着广泛的应用前景。

7.5.2 蚁群算法

群体智能算法是基于群体行为对给定的目标进行寻优的启发式搜索算法，其寻优过程体现了随机、并行和分布式等特点。对于每个智能个体，其定义本身是相对的，其大小和功能要根据所求解的问题而定，并且每个智能个体即使处于合理的寻优进程之中，其个体动态也不能保证在每个时刻都具有最佳的寻优收敛特征。智能寻优方式的实现是在没有集中控制并且不提供全局模型的前提下，通过整个智能群体的总体优化特征来体现的。

- 群体智能（Swarm Intelligence）的优点主要体现在：
- ① 无集中控制约束，控制是分布式的，不会因个体的故障影响整个问题的求解，确保了系统具有更强的鲁棒性；
 - ② 群体中的每个个体都能够改变环境，这是个体之间间接通信的一种方式，群体间以非直接的信息交流方式确保了系统的扩展性，由于系统中个体的增加而增强的通信开销也少；
 - ③ 并行分布式算法模型，可充分利用多处理器，这样的分布模式更适合于网络环境下的工作状态；
 - ④ 对问题定义的连续性无特殊要求；
 - ⑤ 系统中的每个个体的能力十分简单，每个个体的执行时间也比较短，并且算法实现简单；
 - ⑥ 群体具有自组织性，因为群体表现出来的复杂行为是通过简单个体的交互过程凸现出智能的。

作为群体智能算法的典型代表，与蚁群和鸟群典型的行为相对应，研究者分别提出了蚁群优化（Ant Colony Optimization, ACO）算法和粒子群优化（Particle Swarm Optimization, PSO）算法。

- #### 1. 蚁群智慧
- (1) 群居昆虫的特性
- 群居昆虫几乎遍布整个生物圈，群居昆虫的集体行为称为“群集智能”。群集智能的成功在于即使个体遵循的不过是一些简单的规则，最后的群体行为也可能出奇复杂，而且格外有效。之所以如此成功，是因为它们具备以下几个特性：
- 灵活性：群体可以适应随时变化的环境；
 - 稳健性：即使个体失败，整个群体仍然能完成任务；
 - 自我组织：活动既不受中央控制，也不受局部监管。
- (2) 蚁群集体行为的信息正反馈
- 单个蚂蚁的智力水平很低，工作也不受监管。对群居昆虫来说，团队合作在很大程度上是自发组织的，主要通过群居成员之间个体的互动进行协调；尽管每一次互动可能非常简单

(一只蚂蚁只是跟踪另一只蚂蚁留下的痕迹),但整个群体却能解决复杂的问题。例如,从无数条可能的路线中找出接近食物源的最短路线。

蚁群寻找食物时会派出一些蚂蚁分头在四周寻找,如果一只蚂蚁找到食物,它就会返回巢中通知其他的蚂蚁,并且在沿途留下信息素作为蚁群前往食物所在地的标记,蚂蚁个体之间通过这种信息的交流来寻找食物。信息素随时间挥发,在较短的路径上浓度较大。如果两只蚂蚁同时找到同一食物,又采取不同路线返回巢中,那么较远的路径上的信息素的气味会比较淡,而较近的路径上的比较浓的信息素气味将吸引蚁群沿着这一条较近的路线前往食物所在地。这样,由大量蚂蚁组成的蚁群集体行为表现出了一种信息正反馈现象。

2. 蚁群算法的基本思想

基于对蚂蚁在寻找食物过程中发现路径行为的研究,意大利学者 Marco Dorigo 于 1992 年在他的博士论文中使用人工蚂蚁来模仿自然界的蚂蚁,并模仿它们的觅食过程,提出了基本蚁群算法。

设 A 是巢穴, E 是食物源, HC 为一障碍物。由于障碍物存在,蚂蚁要想由 A 到达 E ,或者由 E 返回 A ,只能由 H 或 C 绕过障碍物。各点之间的距离如图 7.55(a)所示。

假定每个时间单位有 30 只蚂蚁由 A 到达 B ,有 30 只蚂蚁由 E 到达 D 点,蚂蚁过后留下的激素物质质量(以下称为信息素)为 1,设该物质停留时间为 1。

在初始时刻,由于路径上均无信息存在,位于 B 和 E 的蚂蚁可以随机选择路径。从统计的角度可以认为它们以相同的概率选择 DH , BC , DH 和 DC ,如图 7.55(b)所示。

在 $t = 1$ 时刻,将有 20 只蚂蚁由 B 和 D 到达 C ,有 10 只蚂蚁由 B 和 D 到达 H ,在路径 BCD 上的信息量是路径 BHD 上信息量的两倍。

随着时间的推移,蚂蚁将会以越来越大的概率选择路径 BCD ,最终完全选择路径 BCD ,如图 7.55(c)所示,从而找到由蚁巢到食物源的最短路径。

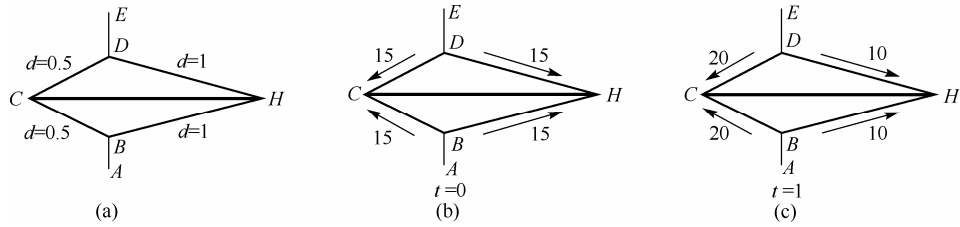


图 7.55 基本蚁群算法

(1) 蚂蚁活动的简单规则

蚂蚁并不需要知道整个世界的信息,它们其实只关心很小范围内的眼前信息,而且根据这些局部信息利用几条简单的规则进行决策,这样,在蚁群这个集体里,复杂性的行为就会凸现出来。

蚁群的高效行为来源于群体活动的蚂蚁个体遵循两条基本规则的约束:释放信息素,跟踪其他蚂蚁留下的痕迹。

通过信息素这个纽带,各个蚂蚁之间关联起来。比如,一只找到了食物的蚂蚁,并没有直接告诉其他蚂蚁这儿有食物,而是向环境播撒信息素,其他蚂蚁经过附近时,就会感觉到信息素的存在,进而根据信息素的指引找到食物。

● 范围

蚂蚁观察到的范围是一个方格世界,蚂蚁有一个参数为速度半径(一般是 3),那么它能观察到的范围就是 3×3 个方格世界,并且能移动的距离也在这个范围之内。

- 环境

蚂蚁所在的环境中，有障碍物，有别的蚂蚁，还有信息素。

信息素有两种：一种是找到食物的蚂蚁撒下的食物信息素，一种是找到窝的蚂蚁撒下的窝的信息素。

每个蚂蚁都仅仅能感知它观察到范围内的环境信息。环境以一定的速率让信息素消失。

- 觅食规则

每只蚂蚁在能感知的范围内寻找是否有食物。如果有，就直接过去；否则看是否有信息素，并且向感知的范围内信息素多的方向走。

每只蚂蚁都会以小概率犯错误，从而并不一定是往信息素最多的点移动。

蚂蚁找窝的规则和上面一样，只不过此时它只对窝的信息素做出反应，而对食物信息素没反应。

- 移动规则

每只蚂蚁都朝信息素最多的方向移动。如果周围没有信息素指引，蚂蚁就按照自己原来运动的方向惯性运动下去，并且运动的方向有随机的小扰动。

蚂蚁会记住最近刚走过了哪些点，如果发现要走的下一点最近已经走过，就会尽量避开，从而防止原地转圈。

- 避障规则

如果蚂蚁要移动的方向有障碍物阻挡，它会随机的选择另一个方向；有信息素指引时，按照觅食规则行动。

- 播撒信息素规则

每只蚂蚁在刚找到食物或者窝时散发的信息素最多。随着它走远的距离，播撒的信息素越来越少。

(2) 蚂蚁如何找到最短路径

在信息素和环境的共同作用下，信息素多的地方会有更多的蚂蚁聚集过来。

假设有两条路从窝通向食物，开始的时候，走这两条路的蚂蚁数量同样多。蚂蚁沿着一条路到达终点以后会马上返回来，这样，走短的路的蚂蚁来回一次的时间就短，即重复的频率快，在单位时间里走过的蚂蚁数目就多，撒下的信息素自然也会多，自然会有更多的蚂蚁被吸引过来，从而撒下更多的信息素……，越来越多的蚂蚁聚集到较短的路径上来，最短的路径就近似找到了。

(3) 蚂蚁简单行为底层规则的特点

- 多样性

多样性源于蚂蚁会犯错误，也就是它会按照一定的概率不往信息素多的地方走，而另辟蹊径。这可以理解为一种创新，这种创新如果能缩短路途，更多的蚂蚁就会被吸引过来。多样性保证了蚂蚁在觅食时不致走进死胡同而陷入无限循环。

- 正反馈

保证了相对优良的信息能够被保存。

可以把多样性看成是一种创造能力，而正反馈是一种学习强化能力。蚁群因此逐渐接近全局最短路径。

多样性过剩，则系统过于活跃，蚂蚁的随机运动过多，蚁群陷入混沌状态；多样性不够，正反馈机制过强，则系统过于僵硬，蚁群不能随着环境变化进行适当的调整。

适应环境的多样性与正反馈的结合，既保证了系统的创新能力，又保证了优良特性能够得到强化。

(4) 参数说明

- 最大信息素：蚂蚁在一开始拥有的信息素总量，该值越大，表示程序在较长一段时间能够存在信息素。
- 信息素消减速度：随着时间的流逝，已经存在于环境的信息素会消减，这个数值越大，消减越快。
- 错误概率：蚂蚁不往信息素最大的区域走的概率，该值越大，则表示这个蚂蚁越有创新性。
- 速度半径：蚂蚁一次能走的最大长度，也表示这个蚂蚁的感知范围。
- 记忆能力：蚂蚁能记住多少个刚刚走过点的坐标，该值越大，则整个系统运行速度越慢，该值越小，则蚂蚁越容易原地转圈，停滞不前。

3. 蚁群算法的应用与发展

作为一种构造型求解方法，蚁群算法非常适用于离散空间的复杂组合优化问题，目前已经成功应用于求解旅行商问题（TSP）、网络路由优化、机器人路径规划、二次规划分配问题，车间生产调度、多处理机调度、资源约束的调度等问题。

针对 ACO 算法进化收敛速度慢而且容易陷入局部最优或者出现停滞现象等缺陷，研究者提出了一些改进方案。例如：

最大最小蚁群系统（Max-Min Ant System, MMAS），这是对 AS 进行完善的一个典型代表，基本思想是仅让每一代中的最好个体所走路径上的信息量进行调整，以加快收敛速度，获得更好的解。为了避免搜索停滞，路径上的信息素浓度被限制在一定的范围内。

将蚁群算法与两交换算法相结合来提高基本蚁群算法的搜索效率。

采用确定性选择和随机选择的选择策略，并动态的调整确定性选择的概率，以提高算法在解空间的完全搜索能力。

利用云模型，统一刻画语言值中大量存在的随机性、模糊性及两者之间的关联性，有效限制蚁群算法陷入局部最优解。

采用遗传算法生成信息素分布，利用蚁群算法求精确解，从而实现优势互补。

习题与思考题

- 1 智能控制的出发点是什么？已经发展了哪些智能控制技术？你认为 21 世纪还将出现哪些新的智能控制技术？
- 2 与传统控制相比，智能控制有什么特点？试列举几种主要的智能控制系统，简述它们的工作原理。
- 3 为什么说目前确定隶属函数的方法还不成熟，隶属函数的确定也不是唯一的？试举出一例说明应当如何确定隶属函数。
- 4 模糊蕴含和模糊推理与传统形式逻辑有什么不同？模糊蕴含具有几种不同的定义？
- 5 试说明模糊控制的结构原理和控制规则，模糊控制器有哪几种设计方法？
- 6 模糊理论还可以应用在哪些领域？试举出一例。
- 7 设计专家控制器时应考虑哪些特点？简述专家控制系统的一般结构模型？
- 8 人工神经网络有哪些特性使它适于控制？实际应用中又有哪些不足？
- 9 有哪几种神经控制器已经得到了实际应用？简述它们的结构和作用原理。
- 10 智能控制已经在哪些领域得到了应用？试举出一例，说明其工作原理和控制性能。

第 8 章 嵌入式操作系统基础

在我们的周围，手持与移动设备（如手机、掌上电脑 PDA）、家庭信息设备（如机顶盒、家庭影院）、车用测控系统、卫星导航 GPS……，形形色色的嵌入式系统已经影响着生活的方方面面。嵌入式系统将与控制对象浑然一体，无处不在，识别出单纯的嵌入式系统将变得越来越难，以至于有专家预言，“嵌入式”这个术语将来会不再使用。

8.1 嵌入式系统

8.1.1 嵌入式系统的基本概念

嵌入式系统（Embedded System）一般被定义为以应用为中心，以计算机技术为基础，软件硬件可裁剪，能满足应用系统对功能、可靠性、成本、体积、功耗严格要求的专用计算机系统。嵌入式系统是将先进的计算机技术、半导体技术、电子技术和各个行业的具体应用相结合后的产物。它可以实现对其他设备的控制、监视或管理等功能；通常由嵌入式处理器、嵌入式外围设备、嵌入式操作系统和嵌入式应用软件等几部分组成。

嵌入式系统是面向用户、面向产品、面向应用的，嵌入式处理器的功耗、体积、成本、可靠性、速度、处理能力、电磁兼容性等方面均受到应用要求的制约，成为半导体厂商之间竞争的热点。嵌入式软件包括操作系统软件与应用软件。近年来蓬勃兴起的嵌入式操作系统已经成为系统软件业的一个热点，其主要指标就是实时性、高效率、安全稳定性、可靠性，结构紧凑，易于剪裁、良好的开放性及高效、易用与可视化的集成开发环境。应用软件是实现嵌入式系统功能的关键，要求固化存储，软件代码质量和可靠性高。

8.1.2 嵌入式系统的体系结构

嵌入式系统作为一类特殊的计算机系统，自底向上一般包括硬件平台、嵌入式操作系统和嵌入式应用几个部分，其体系结构如图8.1所示。

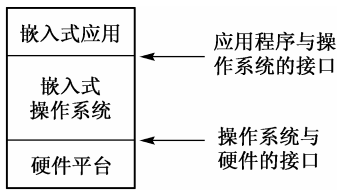


图 8.1 嵌入式系统的体系结构

① 硬件平台是整个嵌入式操作系统和应用程序运行的硬件环境，不同的应用通常有不同的硬件平台。硬件平台的多样性是嵌入式系统的一个主要特点。

嵌入式系统的硬件平台一般由以下模块组成：

- 计算机或者微控制器/微处理器/DSP，字长可能是 4 位、8 位、16 位、32 位，甚至是 64 位；
- 用以保存固化软件（Firmware）的 ROM（只读存储器）；
- 用以存放程序数据的 RAM（随机存取存储器）；
- 连接微控制器/微处理器/DSP 和开关、按钮、传感器、模数转换器、控制器和显示器的 I/O 端口。

越来越多的嵌入式系统以专门的单片微控制器/微处理器/DSP 为核心,并把若干个关键组成部分集成到单个芯片上,体积小而价格便宜,稳定可靠。

② 嵌入式操作系统用来完成嵌入式应用的任务调度和控制等核心功能,具有内核较精简、可配置、与高层应用紧密关联等特点。有些嵌入式系统运行自行编写的前后台程序。

③ 嵌入式应用程序运行于操作系统之上,利用操作系统提供的机制完成特定功能的嵌入式应用。不同的系统需要设计不同的嵌入式应用程序。

8.1.3 嵌入式系统的特点

不同于通用型计算机系统应用的多样性和适用性,嵌入式系统是应用于特定环境下执行面对专业领域的应用系统。与通用计算机相比,它具有以下特点:

① 嵌入式系统通常是面向特定应用的,一般都有实时要求;

② 嵌入式系统是将先进的计算机技术、半导体工艺、电子技术和通信网络技术与各领域的具体应用相结合的产物;

③ 嵌入式系统和具体应用有机地结合在一起,它的升级换代也和具体产品同步进行;

④ 嵌入式系统的硬件和软件都必须高效率地设计,在保证稳定、安全、可靠的基础上量体裁衣,去除冗余,力争在同样的硅片面积上实现更高的性能;

⑤ 为了提高执行速度和系统可靠性,嵌入式系统中的软件一般都固化在存储器芯片或处理器的内部存储器中,而不存储于外部的磁盘等载体中;

⑥ 嵌入式系统本身不具备自举开发能力,用户必须有一套交叉开发工具和环境才能进行开发。

8.1.4 嵌入式系统的类型

嵌入式系统是以嵌入式应用为目的的计算机系统,按照其硬件规模大小可分为

- 系统级:例如,各种类型的工控机、PC104 模块等;
- 模块级:例如,带 CPU 的主板及 OEM 产品等;
- 芯片级:例如,以微控制器、微处理器、DSP、SoC 等为核心的各种产品。

嵌入式系统使用的软件系统决定系统的实时性,按实时性能可分为

- 非实时系统:例如,PDA 等;
- 软实时系统:例如,大多数消费类产品;
- 硬实时系统:例如,有精确严格时间要求的核电站、航空航天器和军工系统等。

实时系统(Real-Time System, RTS)是指能在确定的时间内执行其功能,并对外部的异步事件做出响应的计算机系统。其操作的正确性不仅依赖于逻辑设计的正确程度,而且与这些操作进行的时间有关。“在确定的时间内”是该定义的核心,也就是说,实时系统对响应时间是有严格要求的。

实时系统对逻辑和时序的要求非常严格,如果逻辑和时序出现偏差将会引起严重后果。实时系统有两种类型:软实时系统和硬实时系统。软实时系统仅要求事件响应是实时的,并不要求限定某一任务必须在多长时间内完成;而在硬实时系统中,不仅要求任务响应要实时,而且要求在规定的时间内完成事件的处理。通常,大多数实时系统是两者的结合。实时应用软件的设计一般比非实时应用软件的设计困难,其技术关键是如何保证系统的实时性。

实时系统的软件核心是实时操作系统。实时操作系统(Real-Time Operating System, RTOS)

是指具有实时性、能支持实时控制系统工作的操作系统。其首要任务是调度一切可利用的资源完成实时控制任务，其次才着眼于提高计算机系统的使用效率，重要特点是要满足对时间的限制和要求。实时操作系统具有如下功能：任务管理（多任务和基于优先级的任务调度）、任务间同步和通信（信号量和邮箱等）、存储器优化管理（含 ROM 的管理）、实时时钟服务、中断管理服务。实时操作系统一般的主要特点如下：规模较小，中断被屏蔽的时间短，中断处理时间短，任务切换快。

8.2 嵌入式操作系统

8.2.1 嵌入式操作系统的基本概念

嵌入式操作系统（Embedded Operating System, EOS）是具备相应的存储器管理分配、中断处理、任务间通信和定时器响应，以及提供多任务处理等功能的稳定安全的软件模块集合。是一种支持嵌入式系统应用的操作系统软件，它是嵌入式系统（包括硬、软件系统）极为重要的组成部分，通常包括与硬件相关的底层驱动软件、系统内核、设备驱动接口、通信协议、图形界面、标准化浏览器等。

嵌入式操作系统具有通用操作系统的基本特点，如能够有效管理越来越复杂的系统资源；能够把硬件虚拟化，使得开发人员从繁忙的驱动程序移植和维护中解脱出来；能够提供库函数、驱动程序、工具集及应用程序。与通用操作系统相比较，嵌入式操作系统在系统实时高效性、硬件的相关依赖性、软件固态化及应用的专用性等方面具有较为突出的特点。

一般情况下，嵌入式操作系统可以分为两类：一类是面向控制、通信等领域的实时操作系统，如 Wind River 公司的 VxWorks、ATI 公司的 Nucleus 等；另一类是面向诸如个人数字助理（PDA）、移动电话、机顶盒、电子书等电子产品的非实时操作系统。

具有实时性能的嵌入式操作系统也称为嵌入式实时操作系统（Embedded Real-Time Operating System, ERTOS），它具有实时性强、性能稳定、内核代码短小等特点，用以满足嵌入式实时系统开发的需求。

1. 嵌入式操作系统的发展

伴随着嵌入式系统的发展，嵌入式操作系统经历了四个比较明显的阶段：

(1) 无操作系统的嵌入算法

以单芯片为核心的可编程控制器形式的系统，具有与监测、伺服、指示设备相配合的功能。应用于一些专业性极强的工业控制系统中，通过汇编语言编程对系统进行直接控制，运行结束后清除内存。系统结构和功能都相对单一，处理效率较低，存储容量较小，几乎没有用户接口。

(2) 以嵌入式 CPU 为基础、简单操作系统为核心的嵌入式系统

由于 CPU 种类繁多，通用性比较差；系统开销小，效率高；一般配备系统仿真器，操作系统具有一定的兼容性和扩展性；应用软件较专业，用户界面不够友好；系统主要用来控制系统负载及监控应用程序运行。

(3) 通用的嵌入式实时操作系统

能运行于各种类型的微处理器上，兼容性好；内核精小、效率高，具有高度的模块化和扩展性；具备文件和目录管理、设备支持、多任务、网络支持、图形窗口及用户界面等功能；具有大

量的应用程序接口（API）；嵌入式应用软件丰富。

(4) 基于 Internet 的嵌入式系统

目前，相当数量的嵌入式系统还孤立于 Internet 之外，但随着 Internet 的发展及 Internet 技术与信息家电、工业控制技术等结合日益密切，嵌入式设备与 Internet 的结合正在迅速发展，将代表着嵌入式技术的真正未来。

2. 嵌入式操作系统的特点

嵌入式操作系统 EOS 除具备了通用操作系统最基本的功能，如任务调度、同步机制、中断处理、文件功能等外，一般还具备如下特点：

- ① 可装卸性，开放性、可伸缩性的体系结构。
- ② 强实时性，实时性一般较强，可用于各种设备控制当中。
- ③ 接口统一，提供各种设备驱动接口。
- ④ 操作方便、简单，提供友好的图形 GUI，追求易学易用。
- ⑤ 提供强大的网络功能，支持 TCP/IP 协议及其他协议，提供 TCP/UDP/IP/PPP 协议支持及统一的 MAC 访问层接口，为各种移动计算设备预留接口。
- ⑥ 强稳定性，弱交互性。嵌入式系统一旦开始运行就不需要用户过多的干预，因而要求负责系统管理的 EOS 具有较强的稳定性，嵌入式操作系统的用户接口一般不提供操作命令，它通过系统调用命令向用户程序提供服务。
- ⑦ 固化代码。在嵌入系统中，嵌入式操作系统和应用软件被固化在嵌入式系统计算机的 ROM 中；辅助存储器在嵌入式系统中很少使用。因此，嵌入式操作系统的文件管理功能应该能够很容易地拆卸，而采用各种内存文件系统。
- ⑧ 更好的硬件适应性，也就是良好的移植性。

8.2.2 嵌入式操作系统的实现技术

嵌入式操作系统应当具有多任务和基于优先级的任务调度能力，优化的资源管理、时钟服务和中断服务。

任务调度是实现实时性的关键之一。任务调度算法有优先级调度、轮转调度和时间片调度等。部分嵌入式系统还采用限期（Deadline）调度算法，以保证进程在限定的时间期限内完成任务。

嵌入式系统通常仅有很小的内核，提供应用程序接口的 API 函数，但人机接口可能非常简单，一般不具备完整的系统界面。在内核中提供一般操作系统所应当提供的基本功能，例如，任务调度、内存管理、文件管理、同步机制、中断管理及 API 等。而这些功能可以根据实际系统的需要进行裁减与增加，以最小的代码和资源消耗完成嵌入应用。

嵌入式系统的内核大多有实时性的要求，因此内存的管理比较简单，一般可采用静态内存分配（应用于硬实时调度）、动态内存分配（应用于软实时调度），对于非实时性的任务与进程也可以采用虚拟内存分配算法。

实时性驱动包括时间驱动和事件驱动，采取事件驱动算法较多。中断处理是其关键之一，设计用户中断服务程序时要注意中断现场的保护，以弥补嵌入式系统为追求代码的精简和任务转换的高效率对系统安全性的影响。

1. 嵌入式操作系统的主要技术特性

(1) 内核的占先特性

当系统时间响应很重要时，要使用占先式内核。一旦当前最高优先级的任务就绪，就能立即得到 CPU 的控制权，因而 CPU 的控制权是可知的。使用占先式内核使得任务级响应时间得以最优化。

(2) 调度策略

任务调度策略是直接影响实时性能的因素。强实时系统和准实时系统区别主要在于调度算法的选择。

选择基于优先级调度的算法足以满足准实时系统的要求，而且可以提供高速的响应和大的系统吞吐率；如果两个或两个以上任务有同样优先级，则通常用时间片轮转法进行调度；硬实时系统则需要使用调度方式简单，反应速度快的实时调度算法。

尽管调度算法多种多样，但大多数由单一比率调度算法（RMS）和最早期限优先算法（EDF）变化而来。前者主要用于静态周期任务的调度，后者主要用于动态调度，在不同的系统状态下，两种算法各有优劣。在商业产品中，采用的实际策略常常是各种因素的折中。

(3) 任务优先级分配方法

每个任务都有其优先级。任务越重要，赋予的优先级应越高。应用程序执行过程中各个任务优先级不变，则称为静态优先级。在静态优先级系统中，各个任务及它们的时间约束在程序编译时是已知的。反之，应用程序执行过程中，任务的优先级是可变的，则称为动态优先级。

(4) 时间的可确定性

强实时操作系统的函数调用与服务的执行时间应具有可确定性。系统服务的执行时间不依赖于应用程序任务的多少。系统完成某个确定任务的时间是可预测的。

2. 嵌入式操作系统的功能

与追求性能平均、服务尽可能多及最大吞吐率的通用操作系统不同，嵌入式操作系统着重于改善系统最坏情况下的响应性能，减小中断延迟，以及提供可预测的调度策略。通常应该实现以下几类功能：

(1) 多任务管理及快速的任务切换

包括任务的创建、删除及基于优先级的抢占式调度等。可以实现多任务运行，在需要时甚至可以支持多处理器并行工作。

(2) 任务之间的通信与同步

对于多任务系统，任务之间需要交换数据、同步，以及共享或者排斥资源。

资源排斥是指，在同一时刻只允许一个任务访问某个资源，包括数据结构、代码、硬件资源等。一般可以通过信号量、邮箱、共享内存或者消息队列等来实现。

(3) 迅速的中断响应与中断服务

通过中断服务，快速地响应在系统中异步发生的各种外部事件，并且调度相应的任务运行。

(4) 高精度的时钟控制

管理系统时钟，提供定时和时间服务。通过相应的系统函数调用，应用程序可以获得有关时间的服务。

(5) 高效率的内存管理

提供动态的内存分配机制，增加系统的可靠性，提高了内存使用的效率。

(6) 安全稳定的文件系统管理

为了便于用户进行文档管理，支持相应的文件系统服务，包括文件存取、文件共享及文件打印等。

(7) 方便快捷的网络服务

为用户提供更为高级的各种服务。

3. 嵌入式操作系统的任务调度

调度方式是指，操作系统如何从就绪任务队列中选择一个任务进入运行状态的机制。多任务操作系统按照任务调度方式可以分为抢占式和协作式两类。

抢占式多任务操作系统中，由系统时钟来确定处理器由一个任务转移去执行另外一个任务的时刻，转移时刻是由系统以某种调度策略预先确定的，例如，基于优先级的调度策略，应用程序无法干预操作系统的调度策略。协作式多任务操作系统中，多个应用程序联合协作，通过某种通信方式来共享处理器，但并不是真正意义上的多任务操作系统，难以在多任务操作系统上实现实时应用。因为这种共享方式通常依赖于应用程序主动放弃处理器，只要有一个应用程序一直占着处理器而不放弃，多任务运行就会失败。

抢占式调度的性能比较好，但是需要解决共享代码的可重入性问题，而非抢占式调度相对比较简单，易于扩展。

(1) 嵌入式操作系统的调度算法

如果在系统中存在着一个以上的任务需要获得有保证的响应，系统处理器必须有足够的空闲时间来完成全部工作，而不至于使优先级较低的任务因被抢占而瘫痪。有保证的响应只需要对特定的事件响应得足够快就可以了，而不一定需要达到最快。通常采用以下几种调度算法之一。

● 先来先服务调度算法（First Come First Service Scheduling, FCFSS）

操作系统不强制当前任务退出运行状态，直至当前任务因为某个原因而退出运行状态。这种调度算法非常简单，易于实现，但这种调度算法的缺点是不能保证系统的实时性能。

● 按照时间片循环轮转调度算法（Round-Robin Scheduling）

多个任务平均地取得系统处理器控制权，大多数用于分时操作系统之中，做到公平合理地为各个终端用户服务。这种调度算法也不能保证系统的实时性能。

● 优先级调度算法（Priority First Scheduling）

预先指定各个任务的优先级，在进行调度时从就绪任务队列中选择优先级最高的任务进入运行状态。这种调度算法简单，并且能够迅速地响应高优先级的任务，具有较好的实时性能。缺点是一旦高优先级的任务出现故障，如死循环等，将会影响系统中其他低优先级任务的运行。解决的方法就是通过引入其他机制，如超时错误机制等，来保证一个任务不可能一直独占系统处理器时间。

● 具有时间片循环轮转的优先级调度算法（Priority First with Round-Robin Scheduling）

这就是优先级调度算法加上时间片循环轮转调度算法，当优先权有高低的时候，采用优先级调度算法；若优先权相同，则采用时间片循环轮转调度算法。

(2) 嵌入式操作系统的实时调度机制

实时调度机制是实时操作系统的关键技术之一。调度机制决定在某一时刻，哪一个实时任务可以运行。在实时环境下，考虑到实时任务响应的时限，调度机制必须满足如下基本条件。

- 系统中每一个实时任务都有合理的机会被调度运行；
- 一些优先级更高的实时任务应该获得相对于其他任务更好的运行机会。

对于实时操作系统来说，实时调度机制应该尽可能地减小系统响应时间，以保证实时性能。因此，实时操作系统大多数采用基于优先级的抢占式调度机制。

(3) 实时调度模块的基本功能

实时操作系统的调度模块主要功能是，把当前运行任务从运行状态转换为其他状态，以及把另外一个任务从就绪状态转换为运行状态。

当正在运行的任务 *A* 所占用的处理器时间片到期或者需要等待某个事件的发生（例如，定时时间到，或串行口接收到一个字符）时，调度模块首先把运行任务 *A* 的现场信息保存到 *A* 任务控制块中，并从就绪任务队列中选择一个任务 *B*，恢复保存在 *B* 任务控制块中的现场信息，然后将处理器控制权交给任务 *B*。

在有优先级的调度机制中，如果有任务 *B* 退出等待状态，如所等待的某个事件已经发生，调度模块应该首先把这个退出等待状态的任务 *B* 的优先级与当前运行任务 *A* 的优先级进行比较，如果 *B* 的优先级比 *A* 高，则调度模块应该强制当前运行任务 *A* 从运行状态转换为就绪状态，然后将新的任务 *B* 设置为运行状态。

一个实时调度模块应该完成以下基本功能：

- 当前运行任务与其他任务优先级的比较；
- 保存当前运行任务现场信息；
- 从就绪队列中按照一定的规则选出一个任务进入运行状态；
- 恢复即将进入运行状态的任务的现场信息；
- 将处理器控制权交给新任务。

8.2.3 嵌入式操作系统的性能指标

衡量一个嵌入式操作系统实时性能的主要指标有：

1. 任务切换时间（Task Switching Time）

也称为上下文切换时间，定义为系统在两个独立的、处于就绪态并具有相同优先级的任务之间切换所需要的时间。它包括三个部分，即保存当前任务上下文的时间、调度程序选中新任务的时间和恢复新任务上下文的时间。任务切换时间主要取决于保存任务上下文所用的数据结构及操作系统采用的调度算法的效率。

2. 抢占时间（Preemption Time）

系统将控制从低优先级的任务转移到高优先级任务所花费的时间。为了对任务进行抢占，系统必须首先识别引起高优先级任务就绪的事件，比较两个任务的优先级，最后进行任务的切换，所以抢占时间中包括了任务切换时间。

3. 中断延迟时间（Interrupt Latency Time）

也称为中断响应时间，指从中断到第一条指令所持续的时间间隔。它由四部分组成，即硬件延迟部分（通常可以忽略不计）、RTOS 的关中断时间、处理器完成当前指令的时间，以及中断响应周期的时间。

中断响应时间是系统在最坏情况下响应中断的时间，假如某系统 100 次中有 99 次在 50 ms 之内响应中断，只有一次响应中断的时间是 250 ms，只能认为中断响应时间是 250 ms。另外，还有系统响应时间（系统发出处理要求到系统给出应答信号的时间）、最长关中断时间、非屏蔽中断响应时间等辅助的衡量指标。

4. 信号量混洗时间（Semaphore Shuffling Time）

指从一个任务释放信号量到另一个等待该信号量的任务被激活的时间延迟。在 ERTOS 中,通常有许多任务同时竞争某一共享资源,基于信号量的互斥访问保证了任一时刻只有一个任务能够访问公共资源。信号量混洗时间反映了与互斥有关的时间开销,因此也是衡量 ERTOS 实时性能的一个重要指标。

5. 死锁解除时间 (Deadlock Breaking Time)

即系统解开处于死锁状态的多个任务所需花费的时间。死锁解除时间反映了 RTOS 解决死锁的算法的效率。

6. 数据包吞吐率 (Datagram Throughput Time)

指一个任务通过调用 ERTOS 的原语,把数据传送到另一个任务去时,每秒可以传送的字节数。

8.2.4 嵌入式操作系统的分类

按分类方法可以对嵌入式操作系统进行不同的分类。按是否付费分,可分为需要付费使用的,如嵌入式 Windows, VxWorks 等需要支付许可证使用费的商业操作系统,以及可以免费使用的,如 RTLinux, RTEMS 和 μ C/OS-II 等开源操作系统。按系统实时性能分,可分为非实时、软实时、硬实时等。按内核的占先特性分,可分为使用非占先式内核的嵌入式操作系统和使用占先式内核的嵌入式实时操作系统。

嵌入式实时操作系统的任务主要是多任务的管理,而多任务的管理是通过在嵌入式系统中运行一个实时内核实现的。所以,实时内核是嵌入式操作系统的核心,它对多个任务进行调度、管理,以保证应用程序的实时性。

下面根据嵌入式操作系统内核的占先特性分类进行简单介绍。

1. 使用非占先式内核的嵌入式操作系统

非占先式内核是指每个任务应自我放弃 CPU 的所有权,非占先式调度法也称为协作型多任务,各个任务彼此合作共享一个 CPU。异步事件还是由中断服务来处理。中断服务可以使一个高优先级的任务由挂起状态变为就绪状态。但中断服务以后控制权还是回到原来被中断了的那个任务,直到该任务主动放弃 CPU 的使用权时,那个高优先级的任务才能获得 CPU 的使用权。

非占先式内核有如下优点。

(1) 可以使用不可重入函数

在任务级,非占先式内核允许使用不可重入函数。每个任务都可以调用不可重入性函数,而不必担心其他任务可能正在使用该函数,从而造成数据的破坏。因为每个任务要运行到完成时才释放 CPU 的控制权。当然,被调用的函数本身不得有放弃 CPU 控制权的企图。

使用非占先式内核时,任务级响应时间比前后台系统快得多。此时的任务级响应时间取决于最长的任务执行时间。与占先式内核相比,非占先式内核最大的优点是简单,且占用内存,特别是 RAM 要少得多。

(2) 几乎不需要使用信号量保护共享数据

运行着的任务拥有自己的 CPU,不必担心被别的任务抢走。但这也不是绝对的,在某种情况下,信号量还是用得着的。处理共享 I/O 设备时仍需要使用互斥型信号量,例如,在打印机的使用上,仍需要满足互斥条件。

非占先式内核的最大缺陷在于其响应时间,高优先级的任务已经进入就绪状态,但还不

能运行，也许要等很长时间，直到当前运行着的任务释放 CPU。与前后台系统一样，非占先式内核的任务级响应时间是不确定的，不知道什么时候最高优先级的任务才能拿到 CPU 的控制权，完全取决于应用程序什么时候释放 CPU。

一个没有优先级别的操作系统可以将程序中的后台工作转化为可以预先处理的工作，而不需要将原来的程序分割成多个块。一个不存在优先级别的系统一般相对比较容易执行、调试，并且在设计过程中不存在很多如资源共享的复杂操作。

这类系统的主要不足之处是开发人员必须时刻保护决定将 CPU 转交给其他任务的时间。如果转交时间延迟，即使只有一次，也会影响整个操作系统的响应，导致崩溃。

总之，非占先式内核允许每个任务运行，直到该任务自愿放弃 CPU 的控制权。中断可以打入运行着的任务、中断服务完成以后，将 CPU 控制权还给被中断了的任务。任务级响应时间要大大好于没有使用操作系统的前后台系统，但仍是不可知的，商业软件几乎都不使用非占先式内核。

2. 使用占先式内核的嵌入式操作系统

当系统响应时间很重要时，要使用占先式内核。最高优先级的任务一旦就绪，总能得到 CPU 的控制权。当一个运行着的任务使一个比它优先级高的任务进入就绪状态时，当前任务的 CPU 使用权就被剥夺了，或者说任务被挂起了，那个高优先级的任务立刻得到了 CPU 的控制权。如果是中断服务子程序使一个高优先级的任务进入就绪状态，当中断完成时，中断了的任务被挂起，优先级高的那个任务开始运行。

使用占先式内核使任务级响应时间得以最优化，最高优先级的任务什么时候可以执行，可以得到 CPU 的控制权是可行的。

使用占先式内核时，应用程序不应直接使用不可重入性函数，调用不可重入性函数时，要满足互斥条件，这一点可以用互斥型信号量来实现。如果调用不可重入性函数，低优先级的任务 CPU 的使用权被高优先级任务剥夺，不可重入性函数中的数据有可能被破坏。因此，使用的 C 编译器所提供的原函数应具备可重入性。综上所述，占先式内核总是让就绪态的高优先级的任务先运行，中断服务程序可以抢占 CPU，到中断服务完成时，内核让此时优先级最高的任务运行。

一个完整的具备优先级别的操作系统可以处理包含多个相互竞争的任务和多个软件开发者。具备优先级别的操作系统在外部事件（如中断时间等）和操作系统调用之间转换。根据它们之间相对的优先级别和分配的时间周期自动安排任务，同时减少了任务之间的相互冲突。

然而，在优先级别的系统中的任务转换，需要任务之间的相互通信的外部机制，而且会给调试带来更多困难，比如任务之间的死锁冒险。程序员必须学会使用新的技术和规则来处理这些问题，否则将会导致很多问题和不可靠的代码。

8.2.5 使用嵌入式操作系统的优缺点

嵌入式操作系统在嵌入式系统中的应用越来越广泛，尤其在功能复杂、系统庞大的应用中显得越来越重要。

一般来说，对于开发效率第一、执行效率（包括执行时间和资源占用）第二的应用项目，如果应用项目对额外的需求（嵌入式操作系统的价格、额外 ROM/RAM 开销、CPU 额外负担等）可以承受，应该考虑使用嵌入式操作系统。

1. 使用嵌入式操作系统的优点

① 提高了系统的稳定性和可靠性。在控制系统中，出于安全方面的考虑，要求系统不仅不能崩溃，还要有自愈能力。这要求在硬件设计方面提高系统的可靠性和抗干扰性，在软件设计方面也尽可能地减少安全漏洞和不可靠的隐患。长期以来的前后台系统软件设计在遇到强干扰时，运行的程序会产生异常、出错、跑飞，甚至死循环，从而造成系统的崩溃。而对嵌入式操作系统管理的系统，这种干扰可能只是引起若干进程中的一个破坏，可以通过运行的系统监控进程对其进行修复。通常情况下，这个系统监视进程用来监视各进程运行状况，遇到异常情况时，采取把有问题任务清除掉等有利于系统稳定可靠的措施。

② 实时性好，特别是使用占先式实时内核的操作系统。

③ 充分发挥 CPU 的多任务潜力。嵌入式操作系统是专为多任务操作而设计的，特别适于运行多任务系统。

④ 提高了开发效率，缩短了开发周期。在嵌入式操作系统环境下，开发一个复杂的应用程序，通常可以按照软件工程中的解耦原则将整个程序分解为多个任务模块。每个任务模块的调试、修改几乎不影响其他模块。商业软件一般都提供了良好的多任务调试环境。与目前广泛应用的前后台系统相比，系统的软件维护和功能扩展方便很多。

2. 使用嵌入式操作系统的缺点

① 操作系统本身要占用一定数量的额外内存，包括 ROM 和 RAM。

② CPU 要增加一定的额外负荷，不同的嵌入式操作系统对 CPU 的负荷要求不同。

③ 如果使用商业嵌入式操作系统软件，需要增加额外的成本，而选用免费的嵌入式操作系统，则不一定能得到很好的支持和服务保证。

8.3 典型的嵌入式操作系统

自从嵌入式操作系统诞生以来，针对各种不同的任务和性能需求，许多公司和研究机构甚至个人，设计开发了多种功能和规模各不相同的实时内核和嵌入式操作系统。

目前，比较流行的嵌入式操作系统有数十种，有着很高的适应度和普及性。开发的硬件平台也涵盖了 DSP、ARM、x86、POWER PC 等各种构架的 CPU。著名的商业嵌入式操作系统有 Wind River 公司的 VxWorks、微软公司的 Windows CE、ISI 公司的 pSOS、GNU 组织的 QNX、ATI 公司的 Nucleus PLUS、摩托罗拉、诺基亚、三星等公司的智能手机操作系统 SymbianOS 等，而一些免费的嵌入式操作系统有基于 Linux 的 RTLinux 和 μ CLinux、 μ C/OS-II 等。

8.3.1 嵌入式 Windows

除了在 Windows 桌面操作系统软件方面处于领导地位外，微软也致力于嵌入式 Windows (Windows Embedded) 的开发推广。嵌入式 Windows 是微软开发的嵌入式操作系统软件和工具集系列，可用于各种计算机设备，如收款机、自动柜员机 (ATM)、工业控制器、网关、Internet 协议 (IP) 电话、服务器设备、基于 Windows 的瘦客户机、高级消费类电子产品，以及各种专门的手持设备等。

嵌入式 Windows 产品系列主要包括嵌入式操作系统 Windows CE (Windows Embedded CE)、嵌入式 Windows XP (Windows XP Embedded)，以及用于服务网点 (Point Of Service, POS) 的嵌入式 Windows (Windows Embedded for Point Of Service, WEPOS)，这些操作系统中包含的技术和工具可以开发适用于各种要求严格的嵌入式设备，大大缩短了设备开发的

时间。嵌入式 Windows 产品的应用对象如表 8.1 所示。

表 8.1 根据应用对象选择嵌入式 Windows 操作系统

应用对象	嵌入式 Windows 操作系统		
	Windows Embedded CE	Windows XP Embedded	Windows Embedded for point of service
移动客户端			
PDA	<input type="checkbox"/>		
手持式 GPS 设备	<input type="checkbox"/>		
互联网及媒体应用	<input type="checkbox"/>	<input type="checkbox"/>	
PC 外设			
数码相机	<input type="checkbox"/>		
打印机和扫描仪	<input type="checkbox"/>	<input type="checkbox"/>	
瘦客户端			
基于 Windows 的终端	<input type="checkbox"/>	<input type="checkbox"/>	
服务站			
全功能销售点		<input type="checkbox"/>	<input type="checkbox"/>
移动销售点	<input type="checkbox"/>		
服务亭		<input type="checkbox"/>	<input type="checkbox"/>
微型服务亭	<input type="checkbox"/>		
ATM		<input type="checkbox"/>	<input type="checkbox"/>
加油机	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
可连接客户端			
普通机顶盒	<input type="checkbox"/>		
主端机顶盒	<input type="checkbox"/>	<input type="checkbox"/>	
普通住宅区网关及服务器	<input type="checkbox"/>		
高端住宅区网关及家庭服务器		<input type="checkbox"/>	
工业管理	<input type="checkbox"/>	<input type="checkbox"/>	
Voice Over Internet Protocol (VOIP) 电话	<input type="checkbox"/>		
网络媒体服务器	<input type="checkbox"/>	<input type="checkbox"/>	

另外，Windows Mobile 是基于 Windows CE 构建的一种完善的软件平台，Windows Mobile Smartphone 和 Pocket PC 操作系统专为要求特殊硬件配置的设备而设计。该软件包括标准化的接口和应用程序，可确保在各种硬件设计中的兼容性。

Windows Automotive 是基于 Windows Embedded CE 的实时嵌入式操作系统。在共享 Windows CE 丰富平台和服务的同时，Windows Automotive 还提供语音识别技术、自动化电源管理功能，以及其他自动化特定的选项。

1. Windows Embedded CE

微软于 1996 年 11 月正式进入嵌入式市场，并推出 Windows Embedded CE 这种可自定义的嵌入式操作系统，将可靠的实时功能与高级 Windows 技术集成在一起，是从整体上为有限资源平台设计的多线程、完整优先权、多任务的操作系统。Windows Embedded CE 很多时候也简称为 Windows CE。

Windows CE 的模块化设计允许它对于从掌上电脑到专用的工业控制器的各种用户电子设备进行定制，适用于占用内存很少的设备，是微软专门为非 PC 领域的嵌入式应用设计的战

略性操作系统产品。Windows CE 可以在多种 CPU 上运行，Windows CE 提供 32 位本地硬实时、小型操作系统、统一内核，以及人们熟悉的嵌入式开发工具。包括一个集成的、易于使用的电话用户接口（TUI），一个具有大范围呼叫控制功能的 VoIP 应用程序接口层（VAIL），一个可以访问 Microsoft Exchange 服务器上的联系信息和日历数据的接口，高级配置功能，一个完整的网络层堆栈（便于启用了 VoIP 的设备的开发和基础结构集成）。适合各种小型设备，例如，机顶盒、瘦客户机、数字媒体适配器、IP 语音（VoIP）电话、导航设备、医疗设备、便携式媒体播放器、家庭网关、数字相机、网络数字电视、PDA、GPS 手持设备、各种自动工业设备和高级消费类电子产品，以及配置以下处理器内核的设备：x86, ARM, MIPS 及 SH4。

Windows CE 提供了一系列与 Win32 系统兼容的模块，开发者可以选择利用这些模块作为元件，以及各类可视化的开发工具，例如，Windows CE Tool Kit for VC、VB 和 VJ，来构造自己的产品子系统。

(1) Windows CE 的结构

- 硬件抽象层（OEM 抽象层）OAL。这是硬件与内核之间的接口，OEM 用户接受的是经过剪裁的适用于自己指定系统的功能结构模块。
- 图形窗口与事件子系统。Windows CE 支持 24 位色并且沿用 Win32 的用户接口特征：覆盖式窗口、事件处理、用户接口控制、对话框及内部通信等，并且包含 GUI 和 Windows 格式的位图，因此图形用户界面相当出色。Windows CE 的 API 是 Win32 API 的一个子集，支持近 1500 个 Win32 API。用户可以利用这些 API，编写出任何复杂的应用程序。
- 对象存储程序（Object Store）。包括文件系统、注册表数据库及按需换页、共享存储、交叉处理同步、支持大容量堆栈等高性能、高效率的操作系统特性，提供基于 ROM、基于 RAM 和基于文件分配表 FAT 的文件系统及 Win32 的文件系统功能。为永久性数据和应用提供存储手段，意外的复位与掉电也不会破坏数据，系统重新启动后能够恢复到原来的位置。
- 设备管理和 PC 设备服务。Windows CE 对所有的标准 PC 外部设备提供一个定义好的用户接口（DDI），设计产品时对接口填写参数，运行时通过中断服务程序唤醒驱动程序，由驱动程序完成所有的工作。
- 多种通信手段。提供包括 Win32 的网络功能、串行接口功能和电话拨号应用编程接口（TAPI）及远程访问（RAS）等，广泛支持各种通信硬件，支持直接的局域网连接及拨号连接，并提供与 PC、内部网及 Internet 的连接，包括用于应用级数据传输的设备至设备间的连接。在提供各种基本的通信基础结构的同时，Windows CE 还提供与 Windows 9x/NT 间的最佳集成和通信。
- 类似 Windows 外壳和外壳服务，用于 Windows 界面的系统等。

(2) Windows CE 的技术特点

- 模块化及可伸缩性、实时性能好，通信能力强大，支持多种 CPU，可以使生产厂家与原有的设备混合协调地使用。
- 具有专门为实时嵌入应用而设计的、抢先式多任务的操作系统内核，可以将该内核固化到 ROM 中运行，占用 ROM 和运行所需的 RAM 都很小。
- 可以应用于高速数据采集等嵌入式实时系统，处理中断事件的能力能够满足大部分实时应用的要求，并支持实时时钟和时钟频率计数功能。

- 编码灵活。根据选择的 Microsoft 开发技术，如 Win32、MFC 或 .NET Compact Framework (C#, VB)，可选择本机代码（性能高，规模小）或编写可控代码（速度快，可迁移至其他平台）。
- 源代码访问。借助于共享资源可帮助调试、测试和变更 OS 视图，还可修改和创建不同的功能，同时保持全面控制知识产权。
- 便于配置和测试。Platform Builder 包括支持 Windows CE、基于 ARM 的设备模拟器，便于配置、构建和测试操作系统图像。
- 快速编辑。集成了 IntelliSense 技术、语法高亮，以及位图和注册表文件编辑器。
- 改进的基于 Visual Studio 2005 的新型编译器，增加了 C 语言规范，提高了库的质量，支持 CRT，ATL 和 MFC，改进了运行时安全校验功能。
- 除共享资源外，还包括大量可供选择的量产级驱动器，有助于减少定制硬件环境下，修改驱动器的工作量。
- 支持大量通用应用，包括 Windows Internet Explorer 和 Microsoft Windows Media Player 等。

(3) Windows CE 的应用方案

使 Windows CE 成为特定的工业用嵌入式操作系统的方案有两种：

- 不改动原有的硬件设备，找到一个可以和该硬件配套的已有的 Windows CE 版本，这样只需直接向 Microsoft 公司购买它的使用权，或者请 OEM 厂商针对现有的硬件条件专门开发一个配套的 Windows CE。
- 根据所需要的功能，先选好 Windows CE 目标系统的版本，再订购相应的硬件设备。工业上大量使用时，应向 Microsoft 公司购买该目标系统的使用权。

2. 嵌入式 Windows XP (Windows XP Embedded)

嵌入式 Windows XP 以组件化的形式提供 Microsoft Windows 的完整功能，可以基于 x86 处理器和 PC 架构硬件快速开发可靠的高级嵌入式设备。嵌入式 Windows XP 设备的示例包括零售服务点 (POS) 设备、ATM、基于 Windows 的瘦客户机及高级机顶盒。

嵌入式 Windows XP 是一种专门的操作系统，由各种组件化的 Windows XP Professional 功能和一个开发工具端到端集合组成，用于非个人计算机（非桌面）设备。高级商用和民用设备均可直接运行数千种现有 Windows 应用软件和驱动。

嵌入式 Windows XP 可供设备开发人员利用熟悉的 Windows 环境构建全功能软件，支持各种高级嵌入式设备，例如，ATM、打印机与扫描仪、高级机顶盒、高级住宅网关和家庭服务器、工业机器人、网络媒体服务器、医疗设备、游戏设备。

嵌入式 Windows XP 的技术特点如下：

- 端对端开发工具：端对端开发工具可以优化 OS 定制和构建流程，协助组件设计，加快设备市场投放速度。
- 设计模板：采用支持通用技术和设备的内置设计模板进行设计，包括信息亭、信息应用、机顶盒、家用网关等。
- 集成技术：嵌入式 Windows XP 含有丰富的技术，包括多媒体编解码器和格式、蓝牙、DirectX 和 Microsoft .NET Framework。
- 提高安全性功能：嵌入式 Windows XP 含有 Windows XP Professional 高级安全功能，包括内置防火墙。大部分安全功能便于内置到设备中，保证客户敏感信息的安全。
- 减少端口连接：采用现成的硬件和软件节省时间，包括基于 Win32 的应用、驱动器和服

- 支持嵌入式功能：嵌入式 Windows XP 具有最先进的嵌入式功能，如支持多种引导、存储、部署和管理技术。

3. 用于 POS 的嵌入式 Windows（WEPOS）

WEPOS 基于 Windows XP Embedded Service Pack 2 中的技术，提供了 Microsoft Smarter Retail Initiative，是一种易于使用的零售业最佳服务点平台，对于零售和酒店业、需要连接各种外接设备的用户非常友好。

WEPOS 是微软公司首款专门为垂直市场开发的操作系统。这种操作系统和应用环境具有零售商服务点（POS）系统所需的功能，有利于合作伙伴基于 WEPOS 平台，经济高效地构建自己的 POS 系统并投放市场。

WEPOS 可以节省服务点开发人员的时间和资金，而且是首个可为零售设备外设，提供即插即用支持的服务点操作系统，包括收银机、收据打印机和扫描仪等。

WEPOS 适用于全线服务点系统，例如自动结账系统、信息站、食品订购信息亭、ATM 网络等。

4. 嵌入式 Windows 的优点

借助于嵌入式 Windows 平台配置的专用技术和支持功能，除了加快开发速度，降低成本，还有下列优点：

- 灵活性：只选择有助于缩短开发周期的组件，优化操作系统的尺寸。
- 熟悉的程序：开发人员熟悉 Visual Studio，可以管理本机开发，有助于快速开发嵌入式 Windows 应用和设备。
- 经济高效：仅采用设备所需的组件构建运行时映像，便于设备在低成本硬件环境下运行。
- 技术支持：嵌入式 Windows 产品生命周期支持为 10 年，可以利用微软及其全球合作伙伴社区的大量网络资源、培训和工具。

5. 嵌入式 Windows 的选择

除了可根据应用对象选择相应的嵌入式 Windows 产品外（见表 8.1），也可根据系统软硬件情况选择合适的嵌入式 Windows 操作系统，具体如表 8.2 所示。

表 8.2 根据硬件/软件要求选择嵌入式 Windows

嵌入式 Windows 操作系统		硬件 / 软件要求		
		外处理器	实时操作系统	Win32 程序运行情况
Windows Embedded CE	小型设备用组件化实时操作系统	ARM,MIPS SHx,x86	原生实时支持	自订 Win32 应用程序
Windows XP Embedded	大型设备用组件化实时操作系统	仅 x86	通过第三方插件实现	现有或自订 Win32应用程序
Windows Embedded for Point of Service	针对服务站系统优化的设备专用	仅 x86	通过第三方插件实现	现有或自订 Win32应用程序

8.3.2 嵌入式Linux

源代码公开并且遵循 GPL（General Public License）协议的嵌入式 Linux 是近年来嵌入式操作系统的研究热点。嵌入式 Linux 是按照嵌入式操作系统的要求而设计的一种小型操作系统，由一个很小的内核及一些根据需要进行定制的系统模块组成。其内核一般只有几百 KB 左右，即使加上其他必须的模块和应用程序，所需的存储空间也很小。它有多任务、多进程的系统特征，有些还具有实时性。除了可应用于信息家电（机顶盒、数字电视）、多媒体手机、金融业终端系统等智能数字终端领域以外，嵌入式 Linux 在移动计算平台、电子商务平台、智能工业和商业控制、甚至军事领域都有着广泛的应用前景。

1. 嵌入式 Linux 的优点

- 源代码公开，可以任意修改，以满足自己的应用，并且遵从 GPL，调试也很容易，无须为每例应用交纳许可证费。大量的应用软件可用，其中大部分都遵从 GPL，是开放源代码和免费的，可以稍加修改后应用于用户自己的系统。大量优秀开发工具，且都遵从 GPL，也是开放源代码和免费的。
- 系统稳定性好，在多种架构下可靠运行，层次结构与内核的完全开放。嵌入式 Linux 由很多体积小且性能高的微内核和部件组成。自由软件基金会（Free Software Foundation）的 GNU 计划组为 Linux 开发了许多软件。由于内核代码的完全开放性，内核精悍，运行所需资源少，不同层次和不同领域的用户可以根据具体的应用需求对内核进行改造与剪裁，以较低的成本设计出满足自己要求的嵌入式操作系统。
- 诞生于网络时代，并且带有明显的 UNIX 特性，支持 Internet 所有的网络协议。有一个与 UNIX 相似、以核心为基础、完全内存保护、支持多任务与多进程的操作系统。嵌入式系统的最大的应用前景就是在分布式网络中的第一层，特别是下位机或网络终端。在工业应用领域，可以充分利用它的网络特性，开发分布式应用，并且软件的开发和维护人才较多，成本较低。
- 有一整套的工具链（GCC）。用户不需要专门的仿真器，就能够比较容易地自行建立与配置嵌入式系统，以及在该系统下的开发环境与仿真运行环境，甚至可以在该系统上直接进行系统内核的调试与仿真运行。特别是，Linux 带有用户完全熟悉的 UNIX 完善的开发系统，几乎所有的 UNIX 的应用软件都有 Linux 的相应版本。
- 具有广泛的硬件支持。嵌入式 Linux 和普通 Linux 并无本质区别，PC 上用到的硬件，嵌入式 Linux 几乎都支持，而且各种硬件的驱动程序源代码都可以得到，为用户编写自己专有硬件的驱动程序带来很大的方便。可以支持 RISC 和 CISC 结构的芯片，可以支持 16 位、32 位和 64 位 CPU。就目前的流行器件来说，它可以运行在 Intel 系列 CPU 上，也可以运行在 IBM 和 Motorola 的 Power PC 系列 CPU，以及 Alpha 和 SPARC 系列 CPU 上。
- 灵活性强。编写的应用程序，可以轻而易举地移植到 8 兆内存的嵌入式硬件装置中，并能够可靠和稳定地运行。

2. 嵌入式 Linux 实现技术

将 Linux 作为嵌入式系统应用于微机控制领域时，主要关注的问题有：内核结构与功能

的完善、实时功能的增补、Windows 界面及用户图形接口、集成的开发环境等。比如，图形化的应用环境的选择是采样 X-Windows 还是采用 Mini GUI。

- 标准版本的 Linux 系统内核采用 Monolithic 体系，其优点是模块之间的直接沟通，任务之间的高效切换，系统的响应速度和执行效率高；缺点是体积较大，并且影响系统的启动速度。为此，专门为嵌入式应用编写了基于 Micro Kernel 的微内核，该内核仅仅包括了一些基本的内核功能，如创建和删除任务、任务调度、内存管理和中断处理等，特点是占用资源较少、启动速度快，但是对执行效率有一定的影响。
- 实时功能的开发和扩展。实时性是考核嵌入式系统最重要的指标之一。Linux 是一个通用的操作系统，内核不支持事件优先级和抢占实时性，因此在将 Linux 系统作为嵌入系统应用时，必须开发和扩展实时功能，添加实时软件模块，将 Linux 提升为一个实时操作系统。但是，这些实时软件模块运行的内核空间正是操作系统实现调度策略、硬件中断异常和执行程序的部分。如果代码错误，可能会破坏操作系统，从而影响整个系统的可靠性，这对于实时应用将是一个非常严重的弱点。扩展的内容除支持事件优先级和抢占实时性之外，还包括支持更多的设备和更多的应用范围，如串/并口的数据采集、浮点计算，以及工业级实时网络系统。进一步的开发也包括提升实时任务的优先级（最高级），而将 Linux 本身作为一个优先级较低的任务，实时任务与 Linux 的普通进程之间通过共享内存或 FIFO 的方式交换信息。
- 图形界面的差距。图形用户接口（GUI）的广泛流行是当今计算机技术的重大成就之一，标准版本的 Linux 系统向用户提供了一个图形接口 X-Windows 系统，但是它并不适合嵌入式应用的技术要求。一个图形用户界面通常由显示模型、窗口模型和用户模型三个层次组成，包括以用户为中心、多通道、智能化、高带宽、不限地点和图形编程等技术特征。所涉及的技术包括多线程与多窗口、对话框、消息与消息循环、图形与输入抽象、用户控件、剪切算法、数据结构、多字符集支持及矢量字体等。嵌入式应用对 GUI 的要求是轻型、占用资源少、高性能、可靠性高和可配置。尽管 Linux 为用户提供了一些微型化的 mini-Windows，如 Mini GUI、Micro Windows、Open GUI 及 X-Windows 等，但是就工业环境的开发应用所需的特制图形平台来说，还远未达到用户期望的性能。

对于嵌入式应用而言，完善的集成开发环境是指，包括了编译、连接、调试工具、调试监视器、交叉平台方式下的远程开发，以及具有内核调试功能与跨平台仿真调试功能的可视化的开发工具。目前，Linux 系统向用户提供了 GCC 编译器工具链，该平台包含一个内核开发工具，用户可以使用这个工具完成对内核的剪裁、编译、仿真运行、调试及安装等工作，这一点是其他的操作系统所不具备的。但在基于图形界面的特定系统定制平台的研究方面，与 Windows 系统相比，还存在差距，集成开发环境的建立也需进一步的完善。

3. 典型的嵌入式 Linux 操作系统

最初由赫尔辛基大学的 Linus Torvads 创建的类 UNIX 系统的 Linux，由于开放源代码的特点和嵌入式操作系统的巨大的商业价值，多家公司和研究机构开发出了多款不同类型的嵌入式 Linux 操作系统。比较典型的有：

(1) RTLinux

RTLinux 是由美国墨西哥理工学院开发的嵌入式 Linux 操作系统，到目前为止，已经成功地应用于航天飞机的空间数据采集、科学仪器测控和电影特技图像处理等诸多领域。

RTLinux 开发者并没有针对实时操作系统的特性而重写 Linux 的内核,因为这样做的工作量非常大,而且要保证兼容性也非常困难。为此,RTLinux 提出了精巧的内核,并把标准的 Linux 核心作为实时核心的一个进程,同用户的实时进程一起调度。这样,对 Linux 内核的改动非常小,并且充分利用了 Linux 下现有的丰富的软件资源。

(2) μ CLinux

μ CLinux 是嵌入式 Linux 行业主要厂商之一,Lineo 公司的主打产品,同时也是开放源码的嵌入式 Linux 的典范之作。 μ CLinux 主要是针对目标处理器没有存储管理单元 MMU (Memory Management Unit) 的嵌入式系统而设计的。它已经被成功地移植到了很多平台上。由于没有 MMU,其多任务的实现需要一定技巧。 μ CLinux 是一种优秀的嵌入式 Linux 版本,是 micro-Control-Linux 的缩写。它秉承了标准 Linux 的优良特性,经过各方面的小型化改造,形成了一个高度优化的、代码紧凑的嵌入式 Linux。虽然它的体积很小,却仍然保留了 Linux 的大多数的优点:稳定、良好的移植性、优秀的网络功能、对各种文件系统完备的支持和标准丰富的 API。它专为嵌入式系统做了许多小型化的工作,目前已支持多款 CPU,其编译后目标文件可控制在几百 KB 数量级,并已经被成功地移植到很多平台上。

(3) Embedix

Embedix 也是由 Lineo 推出的,根据嵌入式应用系统的特点,重新设计的 Linux 发行版本。Embedix 提供了超过 25 种的 Linux 系统服务,包括 Web 服务器等。系统需要最小 8 MB 内存,3 MB ROM 或快速闪存。Embedix 基于 Linux 2.2 内核,并已经成功地移植到了 Intelx86 和 PowerPC 处理器系列上。像其他的 Linux 版本一样,Embedix 可以免费获得。Lineo 还发布了另一个重要的软件产品,它可以让在 Windows CE 上运行的程序能够在 Embedix 上运行。Lineo 还将计划推出 Embedix 的开发调试工具包、基于图形界面的浏览器等。可以说,Embedix 是一种完整的嵌入式 Linux 解决方案。

(4) XLinux

XLinux 是由美国网虎公司推出的,号称是世界上最小的嵌入式 Linux 系统,内核只有 143 KB,而且还在不断减小。XLinux 核心采用了“超字节集”专利技术,让 Linux 核心不仅可能与标准字符集相容,还涵盖了 12 个国家和地区的字符集。因此,XLinux 在推广 Linux 的国际应用方面有独特的优势。

(5) PocketLinux

PocketLinux 由 Agenda 公司开发,作为其新产品“VR3 PDA”的嵌入式 Linux 操作系统。它可以提供跨操作系统构造统一的、标准化的和开放的信息通信基础结构,在此结构上实现端到端方案的完整平台。PocketLinux 资源框架开放,使普通的软件结构可以为所有用户提供一致的服务。PocketLinux 平台使用户的视线从设备、平台和网络上移开,由此引发了信息技术新时代的产生。在 PocketLinux 中,称之为用户化信息交换(CIE),也就是提供和访问为每个用户需求而定制的“主题”信息的能力,而不管正在使用的设备是什么。

(6) MidoriLinux

Transmeta公司推出的MidoriLinux操作系统代码开放,在GUN普通公共许可(GPL)下发布,可以在 <http://midori.transmeta.com/>上立即获得。该公司有个名为“MidoriLinux计划”。“MidoriLinux”这个名字来源于日本的“绿色”——Midori,用来反映其Linux操作系统的环保外观。

(7) 红旗嵌入式 Linux

北京中科院红旗软件公司推出的嵌入式 Linux 是国内做得较好的一款嵌入式操作系统。目前,中科院计算所自行开发的开放源码的嵌入式操作系统——Easy Embedded OS (EEOS) 也已经开始进入实用阶段。该款嵌入式操作系统重点支持 p-Java。系统目标一方面是小量化,另一方面能重用 Linux 的驱动和其他模块,有望发展成为功能完善、稳定、可靠的国产嵌入式操作系统平台。

在上面列举的众多嵌入式 Linux 操作系统中,国内对于 RTLinux 和 μ CLinux 研究得较多,很多基于它们的产品已经面世,如华恒公司已经把 μ CLinux 成功移植,并投放到市场。

4. RTLinux 的技术特点

Linux 是一个通用操作系统,将它应用于嵌入式实时环境有许多缺点和不足,特别是在运行内核线程时,Linux 会关闭中断,其他的问题还包括分时的调度,虚拟文件系统的时间不确定性,缺乏高精度的计时等。RTLinux 使用众所周知的虚拟机技术的简单方案来解决上面提到的问题,增加了一个仿真程序来替换 Linux 的底层中断程序。一个小的实时内核与 Linux 内核共享控制处理器,来自硬件的属于实时内核的中断将直接被处理,属于 Linux 内核的中断通过中断仿真程序处理。

如果 Linux 内核中断请求没有被允许,中断模拟程序将在中断队列中标记这次中断的发生。当 Linux 内核的中断请求被允许时,在中断队列中的中断将被执行。因此,实时内核的操作可以得到机器的立即响应,而且 Linux 内核不能延迟实时任务的执行。实时任务与运行在 Linux 内核中的进程之间的通信,通过 FIFOs 以共享内存的方式进行。使用实时内核中的调度器调度实时任务,调度器的算法和策略可以用用户自己定义,这样就保留了 Linux 操作系统所提供的丰富的功能,而且可改动它使其作为一个基本内核与实时内核共享控制 CPU。实际上,系统可以看做具有双内核的操作系统,实时内核拥有更高优先级别的任务,换句话说,基本内核可以看做实时系统的空闲任务,只是在没有实时处理要求的时候运行。

其技术要点如下:

(1) 中断模拟

要在标准 Linux 上增加硬实时能力,首先遇到的一个问题是 Linux 为了达到同步使用关闭中断的方式。混杂在一块的关和开中断操作(x86 处理器的 cli 和 sti 机器指令)造成不可确定的中断分派延迟。Linux 内核是一整块大的内核,在提供系统服务各个部分之间没有一个保护的分界线,要改写 Linux 内核感到非常棘手。因而,在实时 Linux 中,是通过在 Linux 内核与中断控制硬件之间增加一个模拟软件来实现的。当关中断发生时,模拟器的一个变量重新设定,只要中断发生,模拟器将检查这个变量,假如这个值是已经设定的(Linux 中断是允许的),Linux 的中断处理程序将立即调用;如果 Linux 中断是不允许的,则中断处理程序将不会被调用,而是设定一个变量值,并且保存所有挂起的中断的信息。一旦 Linux 中断允许处理时,所有挂起的中断将被处理,这种中断称为软中断。由于 Linux 不能直接控制中断控制器,Linux 的中断不会影响实时中断的处理。

(2) 实时任务

实时任务是一个用户定义的程序,它按照在内核模式下特定的调度方式来执行。最开始的设计是通过 x86 处理器内置的分页机制,给每一个实时任务分配自己的地址空间来进行内存保护。

在每次上下文切换中,页目录是基于寄存器的变化来指向新任务的页目录。任务间的切

换非常频繁，当 TLB 没有命中时，系统在上下文切换时的开销将很大，使系统性能降低；系统的调用在保护模式下也是相当费时的操作。为了提高性能，可以将所有的实时任务运行在内核地址空间，省去了保护模式变换的系统开销。

Linux 一个很有用的特性是：可装载内核模块。内核模块通过内核代码可以动态连接到内核地址空间。每个模块定义了两个例程：`init_module()`和 `cleanup_module()`。可链接模块用来在当前的 RTLinux 中动态创建实时任务。动态链接执行过程中，模块装载解决了符号寻址问题，提高了访问效率。所有任务集中在系统的地址空间，使切换也更简单；一个上下文切换是保存所有整数寄存器到栈中，改变栈的指针指向新的任务；同样也支持有浮点运算的任务。

(3) 实时调度

实时调度器的首要任务是满足所有实时任务的时间要求。在大多数实时系统中，调度器是由复杂的代码块组成，它也不可能扩展到适用任何情况。用户只是通过调节参数来改变调度器的行为，但往往是不够的。一般调度器代码也比较慢。在 RTLinux 中，允许用户编写自己的调度器代码，把它实现为一个可装载的内核模块。因而，可以测试不同的调度策略和算法，找出一个最适合自己应用的调度方式。

(4) 计时

精确的计时是正确的调度器操作必需的。调度器常常要求在一个特定的时刻进行任务切换，计时的错误将引起背离计划的调度，导致影响应用的任务释放抖动。

(5) 实时进程

Linux 中加进了 POSIX 实时扩展部分，引进了实时进程的概念，允许一个进程定义为一个实时进程。Linux 区分实时进程和普通进程，采用不同的调度策略，即先来先服务调度（`SCHED_FIFO`）和时间片轮转调度（`SCHED_RR`）。在 `SCHED_RR` 调度中，任务时间片一旦用完就被移动到优先级队列的队尾，并允许同一优先级的其他任务运行。如果同一优先级没有其他任务，该任务继续运行下一个时间片。`SCHED_FIFO` 是运行直至阻塞的策略。`SCHED_FIFO` 任务按优先级调度，一旦开始就一直运行到结束或阻塞在某种资源上。这不像 `SCHED_RR` 任务那样可共享处理器，另外还有计时器的精度问题。以前提供给用户进程警报信号和 `sleep()` 系统调用只有 1 秒的精度，如此粗糙的计时精度是不适合大多数的实时进程的。当前的版本提供了更高精度的时间间隔，然而，内在的时钟限制了计时的正确性。大部分的 Linux 的核心进程是不能中断的。换句话说，一旦一个进程进入到核心模式，它将运行到系统调用的完成或者被阻塞为止。假如在这期间有一个更高优先级的实时进程准备好运行了，它将不得不等待。由于不需要考虑内核重入的问题，这种设计的方式使内核的开发更为简单。

(6) 可支持的目标结构

RTLinux 支持 Linux 支持的大多数 CPU 和操作平台。目前，RTLinux 支持 x86, PowerPC, Alpha 和 MIPS 等处理器。

(7) 开发环境

RTLinux 通常利用大家熟知的 GNU 工具链进行开发。另外，RTLinux 核也提供对实时模块的调试。

8.3.3 VxWorks

VxWorks 操作系统是美国 WindRiver 公司于 1983 年设计开发的一款嵌入式实时操作系统，是 Tornado 嵌入式开发环境的关键组成部分。它支持多种处理器，如 x86, i960, Sun Sparc,

Motorola MC68xxx, MIPS RX000, POWER PC 等。大多数的 VxWorks API 是专有的, 采用 GNU 的编译和调试器。具有良好的持续发展能力、高性能的内核及友好的用户开发环境。它经过广泛的验证, 以良好的可靠性和卓越的实时性被广泛应用在通信、医疗、军事、航空和航天等高精尖技术和实时性要求极高的领域中, 如卫星通信、军事演习、弹道制导、飞机导航等。在美国的 F-16 战斗机、FA-18 战斗机、B-2 隐形轰炸机、爱国者导弹, 甚至火星探测器上都用到了 VxWorks。

VxWorks 具有可裁剪微内核结构和可以根据需要进行定制的系统模块, 高效的任务管理和灵活的任务间通信, 微秒级的中断处理, 不失实时、多任务的系统特征, 支持 POSIX 1003.1b 实时扩展标准, 支持多种物理介质及标准的、完整的 TCP/IP 网络协议等优秀的特性。然而, 由于操作系统本身及开发环境都是专有的, 价格一般都比较高。一般不提供源代码, 只提供二进制代码。由于是专用操作系统, 需要专门的技术人员掌握开发技术和维护, 所以软件的开发和维护成本都较高, 支持的硬件数量也有限制。

VxWorks 的主要技术特点:

1. 多任务调度

包括两种方式: 优先抢占和轮转调度。

优先抢占: 每一个任务都有一个优先级, 系统核心保证优先级最高的任务运行于 CPU。如果有任务优先级高于当前的任务优先级, 系统立刻保存当前任务的上下文, 切换到优先级高的上下文。

轮转调度: 所有相同优先级, 状态为 ready 的任务公平分享 CPU (分配一定的时间间隔, 使每个任务轮流享有 CPU)。系统有 256 个优先级, 从 0 到 255, 0 为最高, 255 位最低, 任务在被创建时就设定了优先级。

2. 任务间的通信

现代实时系统是建立在多任务间通信的概念上的, 一个多任务环境可将实时应用程序构造为一种独立任务的集合。每一个任务都有其单独的执行线程和自己的系统资源集合。任务间通信应使这些任务同步并且协调它们的活动。

3. POSIX 接口

POSIX (可移植操作系统接口), 属于遵照 ISO/IEEE 章程的软件开发标准。这项标准的目的在于支持类 UNIX 操作系统应用程序的可移植性。尽管这些标准仍在研究中, 但已经产生出一系列的接口 (POSIX 标准 1003.1b) 用在重要的实时操作系统服务上。应用这些接口可以比较容易地将应用程序从一个操作系统移到另一个操作系统。几乎所有 POSIX1003.1b 接口都适合于 VxWorks, 包括异步 I/O、信号量、消息队列、存储管理、队列信号、调度、时钟和同步器等。

4. 中断处理

VxWorks 支持用于处理硬件中断和软件陷阱的程序, 而不必使用汇编语言代码。C 程序库为连接到硬件中断向量和控制处理器中断提供相应的子程序。

5. 存储管理

为了保证嵌入式系统的实时性, 要求内存分配过程尽可能地迅速。因此, 在嵌入式系统中, 不可能采用通用操作系统中复杂而完善的内存分配策略, 一般都采用简单、快速的内存分配方案。对实时性要求不同的程序, 分配方案也有所不同。如果系统对于实时性和可靠性

的要求极高，不能容忍丝毫延时或者一次分配失败，当然需要采用静态分配策略，也就是在程序编译时所需要的内存都已经分配好了。

静态分配方案必然导致很大的浪费，因为内存分配必须按照最坏情况进行最大的配置，而实际上在运行中可能使用的只是其中的一小部分。而且在硬件平台不变的情况下，不可能灵活地为系统添加功能，从而使得系统的升级变得困难。

VxWorks 对内存的使用不是段页式的管理，而采用的是分区管理扁平模式（Flat Mode）：静态分配的 IMAGE 占用空间（.code, .data, .bss），系统的动态空间（wind kernel 创建的 REGION#0）和用户的动态空间。

6. 文件系统

VxWorks 提供的快速文件系统适合于实时系统应用。它包括几种支持使用块设备（如磁盘）的本地文件系统。这些设备都使用一个标准的接口，从而使得文件系统能够被灵活地在设备驱动程序上移植。VxWorks 也支持 SCSI 磁带设备的本地文件系统。VxWorks I/O 体系结构甚至还支持在一个单独的 VxWorks 系统上同时并存几个不同的文件系统。VxWorks 支持的文件系统有 dosFs, rt11Fs, rawFs, tapeFs, nfs, cdromfs 等。另一方面，普通数据文件，外部设备都统一作为文件处理。它们在用户面前有相同的语法定义，使用相同的保护机制。这样，既简化了系统设计，又便于用户使用。

7. 可支持目标结构

如 x86, 68k, PPC, CPU 32, i960, SPARC, SH, ColdFire, R3000, R4000, C16X, ARM, MIPS 等。

8. 开发环境

通过板卡支持包（Board Support Package, BSP）支持广泛的实验板和嵌入式计算机平台，已开发的 BSP 支持大量的厂商，如 Motorola, Force, Intel/Cyclone 和 Digital 等。

8.3.4 Nucleus PLUS

Nucleus 是 ATI（Accelerated Technology Inc.）公司于 1990 年推出的实时多任务操作系统，经不断完善为目前的 Nucleus PLUS 版本。Nucleus PLUS 是为实时嵌入式应用而设计的一个抢先式多任务操作系统内核，其 95% 的代码是用 ANSI C 写成的，便于移植并能够支持大多数类型的处理器。从实现角度来看，Nucleus PLUS 是一组 C 函数库，应用程序代码与核心函数库连接在一起，生成一个目标代码，下载到目标板的 RAM 中或直接烧录到目标板的 ROM 中执行。在典型的目标环境中，Nucleus PLUS 核心代码区一般不超过 20 KB 大小。

Nucleus PLUS 采用了软件组件的方法。每个组件具有单一而明确的目的，通常由几个 C 及汇编语言模块构成，提供清晰的外部接口，对组件的引用就是通过这些接口完成的。除了一些特殊情况外，不允许从外部对组件内的全局进行访问。由于采用了软件组件的方法，Nucleus PLUS 各个组件非常易于替换和复用。

Nucleus PLUS 的组件包括任务控制、内存管理、任务间通信、任务的同步与互斥、中断管理、定时器及 I/O 驱动等。

Nucleus 实时多任务操作系统提供源代码给用户。这在技术方面给用户极大的方便，即无须编写和调试 BSP，从而达到易学易用的目的，加速产品上市。

Nucleus 的调试基于全新的动态连接库（DLL）。用户只需通过监控程序或者 BDM 调试口（或者 JTAG 调试口）建立目标系统与主机之间的通信，并给主机上的调试器初始化特性中加入 Nucleus 的动态连接库，这时，调试器就能够自动地去识别运行在目标系统中的 Nucleus

内核和各个应用任务，从而完成任务级调试。

另外，由于有了源程序，用户在调试程序时可以清楚地通过调试命令，追踪到 RTOS 的内层中去，观看和学习内核对任务的管理和调度机制。对于研究 RTOS 深层技术的工程师来说极为方便。

Nucleus PLUS 具有如下特点：

1. 源代码开放

Nucleus PLUS 提供注释严格的 C 源级代码给每一个用户。这样，用户能够深入地了解底层内核的运作方式，并可根据自己的特殊要求，删减或改动系统软件，这对软件的规范化管理及系统软件的测试都有极大的帮助。另外，由于提供了 RTOS 的源级代码，用户不但可以进行 RTOS 的学习和研究，而且产品在量产时也不必支付许可使用费，可以省去大量的费用。对于军方来说，由于提供了源代码，用户完全可以控制内核，而不必担心操作系统中可能会存在异常任务，从而导致系统崩溃。

2. 性价比高

Nucleus PLUS 由于采用了先进的微内核（Micro-kernel）技术，因而在优先级安排、任务调度、任务切换等各个方面都有相当大的优势。另外，对 C++ 语言的全面支持又使得 Nucleus PLUS 的内核成为名副其实的面向对象的实时操作系统内核。而且价格比较容易接受。

3. 易学易用

Nucleus PLUS 能够结合 Paradigm, SDS 以及 ATI 自己的多任务调试器组成功能强大的集成开发环境，配合相应的编译器和动态连接库各类底层驱动软件，用户可以轻松地进行 RTOS 的开发和调试。另外，由于这些集成开发环境为所有的开发工程师所熟悉，因而，容易学习和使用。

4. 功能模块丰富

Nucleus PLUS 除提供功能强大的内核操作系统外，还提供种类丰富的功能模块。例如，用于通信系统的局域网和广域网模块，支持图形应用的实时化 Windows 模块，支持 Internet 网的 WEB 产品模块，工控机实时 BIOS 模块，图形化用户接口，以及应用软件性能分析模块等。用户可以根据自己的应用来选择不同的应用模块。针对各种嵌入式应用，Nucleus PLUS 还提供相应的网络协议（如 TCP/IP, SNMP 等），以满足用户对通信系统的开发要求。另外，可重入的文件系统、可重入的 C 函数库及图形化界面等，也给开发者提供了方便。

5. 支持多款 CPU 类型

Nucleus PLUS 的 RTOS 内核可支持如下类型的 CPU：x86, 68xxx, 68HCxx, NEC V25, ColdFire, 29K, i960, MIPS, SPARClike, TI DSP, ARM6/7, StrongARM, H8/300H, SH1/2/3, PowerPC, V8xx, Panasonic MN10200, Tricore, Mcore 等。

对于 Motorola 68K 和 PowerPC，用户可以利用 GreenHill 公司的 Multi 调试器或 TRACE-ICD 来完成 Nucleus PLUS 的调试；对于 ARM 和 StrongARM，用户可以利用 ARM 公司的 SDT251 调试器或 TRACE-ICD 来完成 Nucleus PLUS 的调试；对于 Intel x86 实模式，用户可以利用 Paradigm 公司的调试器及其 DLL 来进行任务级调试；对于 Intel x86 保护模式，则可以利用 SSI 公司 Softprobe 调试器和 SSI DLL 来调试；对于 i960, SH3/4, ARM6/7, MIPS 等芯片，则可以使用 ATI 公司自己的 UDB 调试器来进行任务的调试。

6. 强大的开发工具

针对不同的 CPU 类型，Nucleus 提供编译器、动态连接库、多任务调试器等相应的工具

来配置用户的开发环境。值得提出的是，ATI 公司最近还发表了基于 Microsoft Developers Studio 的嵌入式集成开发环境——NUCLEUS EDE。从而，率先将嵌入式开发工具与 Microsoft 的强大开发环境结合起来，提供给工程师们强大的开发手段。

Nucleus 在国内外拥有众多客户，其应用覆盖了网络、路由、桥接、Hubs、数据通信、机顶盒、数字摄像机、ISDN、调制器、数字绘图仪、GSM、蜂窝电话、PDA、打印机、GPS、无线通信、汽车、医疗仪器、RAID、适配卡、智能卡、安全保密、工业控制、扫描仪、煤气分析仪、游戏机、多媒体、手持产品、消费产品、局域网、广域网、导航设备、卫星通信、自动提款机、视频产品、条码机、过程控制等诸多领域。

8.3.5 RTEMS

多处理器实时操作系统（Real Time Executive for Multiprocessor System，RTEMS）诞生于 20 世纪 80 年代，是一个纯粹的 32 位系统，由 OAR 公司开发和维护，最早用于美国国防部的导弹控制系统，当时 RTEMS 的全称是“实时导弹控制执行体”。自诞生之时，它就是针对嵌入式系统设计的实时操作系统，以高实时性和高可用性为标准，并且采用了面向对象技术、构件技术等到现在也非常先进的理念构造而成，具有良好的实时性和稳定性。

经过了 20 多年的持续开发，RTEMS 拥有非常广泛的客户，也具有丰富的开发资源。尤其在实时性、稳定性、开发速度和多处理器支持方面都非常优秀，现已逐渐应用于军事、工业和科学工程中。加速器领域广泛使用的控制系统软件 EPICS 也已经被移植到了 RTEMS 操作系统中。

RTEMS 各种系统组件被分割成独立的模块，而这些模块可以在需要时和应用程序直接连接起来。该系统也可以通过选择合适的结构参数进一步裁剪，以满足应用程序的特别需要。一个典型的 RTEMS 应用是通过编译这个应用本身，并和应用程序链接在一起来创建的，然后，创建一个可执行程序，下载到目标系统或烧制到 ROM。所有的系统服务和库对任何应用都是直接可用的。

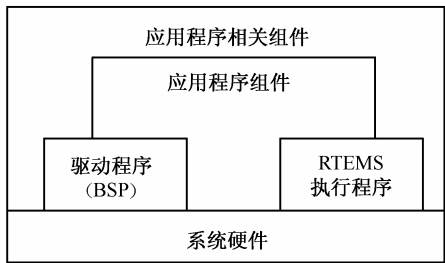


图 8.2 RTEMS 整体结构

RTEMS 的整体结构如图8.2所示。

RTEMS 的主要技术特点如下：

1. 任务调度

RTEMS 任务调度算法保证了在系统内任何一个处理器节点上运行的任务都是处于就绪（Ready）状态中优先级最高的那个任务。

(1) 以优先级为基础

任务调度的控制中，最重要和最有效的是允许用户在任务创建的时候为其分配一个优先级，同时也允许用户在任务运行时改变任务的优先级。RTEMS 提供从 1 到 255 依次降低的 255 个优先等级。当一个任务进入就绪状态时，它前面应该是具有相同或者更高的优先级的任务。该规则提供一个轮转的调度方式，在一组具有相等的优先级任务中，根据它们准备好的次序依次调入 CPU 中运行。虽然用户可以动态调整任务的优先级，但是 RTEMS 调度程序总是选择准备态中的具有最高优先级任务进入 CPU 运行。

(2) 设置抢占位

用户改变调度的一种方法是动态调整任务的优先级，另一种方法是改变正在执行的任务的抢占模态标志（RTEMS_PREEMPT_MASK）。如果将一个任务的抢占位置为无效（RTEMS_

NO_PREEMPT)，则该任务在运行时不会被即使是较高优先级的任务抢占，直到任务结束或者被阻塞。需要注意的是，抢占位设置对于等待调度的任务无效。

(3) 在同等优先级条件下采用时间片轮转调度

基于优先级的时分复用，也称为轮转调度，是一种使用时分复用模态标志（RTEMS_TIMESLICE_MASK）的优先级调度附加算法。如果时分复用状态位置为有效，RTEMS 将会限制该任务在处理器中的运行时间，一旦属于它的运行时间片用完，即被其他任务取代；如果当前优先级就绪队列中没有其他任务，RTEMS 将会增加当前任务的时间片并让其继续执行。必须注意的是，一个较高的优先级任务就绪，就会立刻抢占正在时间片轮转的任务，即使正在运行的任务还没有用完它的整个时间片。

2. 中断管理

当中断产生，RTEMS 将保存处理器的寄存器，然后转向中断处理程序（Interrupt Service Routine, ISR）。ISR 负责处理中断，并操作外部设备。

为了尽量减少对低优先级中断的屏蔽，与 ISR 关系不大的处理尽可能放在应用层完成，从而使 ISR 精简高效。当 ISR 结束时，处理器的控制权将交还给 RTEMS 中断管理器，由中断管理器负责被中断任务的恢复工作。许多处理器支持不同等级的中断（或称为中断优先级），在 RTEMS 中，逻辑上支持 256 个中断等级，这些中断等级被映射到处理器的实际的中断等级上。

在执行系统调用时，有些关键代码是不应该被中断的。当进入这些代码段时，需要尽可能的屏蔽外部中断，当关键代码执行完后，RTEMS 恢复外部中断。RTEMS 优化了中断屏蔽和恢复代码，使得屏蔽和恢复中断的消耗时间几乎可以忽略。

3. 时钟与定时器

时钟管理器提供对日期和其他与时间相关的服务，系统用硬件的实时时钟（RTC）产生周期型的时钟中断。时钟 ISR 会调用 rtems_clock_tick 函数通知系统已经经过了一个特定的时间段。在 RTEMS 中，时间的单位是“嘀哒”（Tick）。

4. 任务间通信与同步

信号量管理器利用信号量管理算法实现信号量，为任务提供同步和互斥。

消息管理器提供任务间通信和使用 RTEMS 消息队列同步的能力。消息是一个可变长度的缓冲区，用于存储任务间通信的信息。消息的长度由用户定义，可以是真实的数据、指针或为空。消息队列允许在任务或者 ISR 之间传递信息。消息队列能包含零个或者多个消息。事件管理器提供一个任务间通信和同步的高性能方法。事件标志被任务（或 ISR）用来通知另一个任务有事件发生。每个任务可以有 32 个相关的事件标志。一个或多个事件标志的集合称为事件集合。

信号管理器为任务提供异步通信的手段。信号管理器允许一个任务定义一个异步信号例程（Asynchronous Signal Routine, ASR）。ASR 和 ISR 很接近。当处理器被中断，应用的执行转到 ISR。同样地，当一个信号被发送到一个任务时，该任务的执行将会转到 ASR。送一个信号给一个任务不会影响任务的状态。信号被任务（或 ISR）用来告知另一个任务有等待的情形发生。每个任务有 32 个相关信号标志。

5. RTEMS 内存管理

定长内存区管理器为动态分配固定尺寸的内存单元提供了接口。定长内存区是一个连续的内存块，内存块中有固定大小的缓冲区。使用链表的方式管理和维护缓冲区定长内存区。

任务从定长内存区内存队列的头部获取缓冲区，释放时放回队列的尾部。与平时常用的指针链表略有不同，RTEMS 中链表的组织使用缓冲区中特定的 8 个字节作为链表指针。缓冲区分配后，整个缓冲区都可以用来存储数据。由于没有专门的 MMU 模块，如果因为错误操作修改了缓冲区附近的内存区域，则可能导致链表及数据的破坏，所以对内存定长内存区的操作应该严格使用管理器提供的 API。

变长内存管理器提供 API 为应用分配变长的内存区。变长内存 (region) 是一个连续内存区，区内有用户定义的变长内存块 (页面)。页面大小可变，但是必须大于 4，并且是 4 的整数倍。如果需求的内存大小大于页面大小，那么页面将自动增加一倍。例如，需要 350 字节内存，则 256 字节的页面将自动扩大到 512 字节。

变长内存管理器将变长内存区按照双向链表组织。对内存请求使用首次匹配算法。为了系统的执行效率，任务在请求了内存后，最好不要频繁地修改获得内存的尺寸，也不应该改变变长区域以外的内存块尺寸。释放的内存块会和邻接的内存块合并，以形成较大的空闲区。

6. 单调周期调度管理

单调周期任务指的是周期性执行的任务。在 RTEMS 中，单调周期管理器提供了执行单调周期任务的 API。周期性任务必须在固定的时间间隔启动。系统启动的时间间隔称为周期。周期性的两个重要指标特征是：执行周期和每次执行的时间，可以用来衡量任务占用处理器的比率。

7. 异常与自定义扩展

在 RTEMS 中，通常将异常称为致命错误 (fatal error)。异常管理器用于处理所有的无法恢复的错误。如果 RTEMS 发现异常，RTEMS 将自动唤醒异常管理器。同样，如果是用户发现异常，那么用户应当唤醒异常管理器。

每种状态码或动态的用户扩展集都可能包括一个异常处理句柄 (函数)。静态扩展集发现异常时可以和宿主机上的调试器通信。如果用户定义了异常处理函数，那么将在出现异常时被异常管理器唤醒；如果用户没有定义异常处理函数或者用户在异常处理函数中将控制权回传给 RTEMS 异常处理器，那么将会调用默认的异常处理函数，此时系统状态将会是 “failed”。

8. 系统移植

应用 RTEMS I/O 管理器需要给出设备驱动表及配置表格的地址。该表在初始化期间由 RTEMS 初始化，它包含每个驱动程序的功能函数入口。

板支持包 (BSP) 包括硬件初始化代码、设备驱动程序、用户扩展、多处理器接口等，用于让 RTEMS 及应用程序访问对应的硬件平台。这些软件当处理器被复位时，基于 RTEMS 的应用就会重新初始化。初始化代码负责初始化硬件平台的各个器件。在不同的硬件平台上，虽然初始化代码不同，但逻辑功能大同小异。

RTEMS 的一个重要设计目标是超越目标硬件的硬件边界，为此向应用软件提供了一个目标硬件系统的逻辑视图，其中各个处理器是透明的。应用软件开发人员可以将包括任务、队列、事件、消息、信号量和内存块这样一些对象设计为全局对象。各个处理器节点上的任务可以访问这些全局变量。RTEMS 自动判断被存取的对象处于本地节点上，还是处于其他处理器上。简单地说，RTEMS 允许整个多处理器系统的软硬件在逻辑上表现为一个系统。

9. 可支持目标结构

RTEMS 设计为移植性强的操作系统，因此，它能支持许多种 CPU 结构，比如，m68k, ColdFire, Hitachi SH, Intel i386, Intel i960, PowePC, SPARC, AMD A29k 和 HP PA-RISC 等。

10. 开发环境

RTEMS 通常利用 GNU 工具链进行开发。

8.3.6 μ C/OS-II

μ C/OS-II 是 Jean J. Labrosse 开发的一个高度可移植的实时操作系统，专为嵌入式应用设计。 μ C/OS-II 读做“microCOS 2”，意为“微控制器操作系统版本 2”。自从 1992 年发布以来，在各个领域得到了广泛的使用，例如，照相机行业、医疗器械、音响设施、发动机控制、网络设备、高速公路电话系统、自动提款机、工业机器人，等等。很多高等院校将 μ C/OS 用于实时系统教学。

μ C/OS-II 是一个完整的，可移植、固化、裁剪的占先式实时多任务内核，实际对象是嵌入式系统。它用 ANSI 的 C 语言编写，包含一小部分汇编语言代码，可用于各类 8 位、16 位和 32 位等超过几十种不同架构的微控制器或 DSP。

μ C/OS-II 的主要特点：

1. 源代码开放

μ C/OS-II 的全部源代码已经公开。源码清晰易读，且结构协调、注解详尽、组织有序。

2. 可移植性

μ C/OS-II 源码绝大部分是用移植性很强的 ANSI C 编写的，与微处理器硬件相关的部分是用汇编语言编写的。汇编语言写的部分已经压到最低限度，以使 μ C/OS-II 便于移植到其他微控制器上。 μ C/OS-II 可以移植的条件是，该微控制器具有堆栈指针，具有 CPU 内部寄存器入栈、出栈指令。另外，使用的 C 编译器必须支持内嵌汇编 (Inline assembly)，或者该 C 语言可扩展和可链接汇编模块，使得关中断和开中断能在 C 语言程序中实现。 μ C/OS-II 可以在绝大多数 8 位、16 位、32 位和 64 位微处理器、微控制器及数字信号处理器上运行。

应用程序从 μ C/OS 升级到 μ C/OS-II 几乎不需要改动，移植的范例可从互联网上找到，网址是 <http://www.ucos-II.com>。

3. 可固化

μ C/OS-II 是为嵌入式应用而设计的，这就意味着，只要具备合适的系列软件工具 (C 编译、汇编、链接及下载/固化)，就可以将 μ C/OS-II 嵌入到产品中，作为产品的一部分。

4. 可裁剪

可以只使用 μ C/OS-II 中应用程序需要的系统服务。也就是说，某产品可以只使用很少几个 μ C/OS-II 调用，而另一个产品则可能使用了几乎所有 μ C/OS-II 的功能，这样可以减少占用的存储器空间 (RAM 和 ROM)。可剪裁性是靠条件编译实现的，只要在用户的应用程序中定义 μ C/OS-II 中的功能哪些是应用程序需要的就可以了。

5. 可占先性

μ C/OS-II 是完全占先式的实时内核，即总是运行就绪条件下优先级最高的任务，性能上与大多数同为占先式的商业内核类似。

6. 多任务

μ C/OS-II 可以管理 64 个任务，除保留 8 个外，留给用户的应用程序最多可有 56 个任务。赋予每个任务的优先级必须是不相同的，因而 μ C/OS-II 不适用于调度优先级平等的时间片轮转调度法 (Round-robin Scheduling)。

7. 可确定性

绝大多数 μ C/OS-II 的函数调用和服务的执行时间具有可确定性，用户总可以知道 μ C/OS-II

的函数调用与服务执行了多长时间。除了函数 `OSTime_Tick()` 和某些事件标志服务， $\mu\text{C}/\text{OS-II}$ 系统服务的执行时间不依赖于用户应用程序任务数目的多少。

8. 任务栈

每个任务都有自己单独的栈。 $\mu\text{C}/\text{OS-II}$ 允许每个任务有不同的栈空间，以便减少应用程序对 RAM 的需求。使用栈空间校验函数，可以确定每个任务到底需要多少栈空间。

9. 系统服务

$\mu\text{C}/\text{OS-II}$ 提供的系统服务包括信号量、互斥型信号量、事件标志、消息邮箱、消息队列、块大小固定的内存的申请与释放，以及时间管理函数等。

10. 中断管理

中断可以使正在执行的任务暂时挂起。如果优先级更高的任务被该中断唤醒，则高优先级的任务在中断嵌套全部退出后立即执行，中断嵌套层数可达 255 层。

11. 稳定性与可靠性

该系统自从 2.51 版本之后，就通过了美国联邦航空管理局（Federal Aviation Administration, FAA）的认证，可以运行在诸如航天器等对安全要求极为苛刻的系统之上。 $\mu\text{C}/\text{OS-II}$ 在一个航空项目中得到了美国对用于商用飞机的符合 RTCA DO-178B 标准的认证，这一结论表明， $\mu\text{C}/\text{OS-II}$ 的每一种功能、每一个函数及每一行代码都经过了考验与测试，具有足够的安全性及稳定性。

对于嵌入式系统而言，选择可以免费获得代码的 $\mu\text{C}/\text{OS-II}$ 是一种经济的选择。

8.4 嵌入式操作系统应用

8.4.1 嵌入式操作系统的选择

在设计开发嵌入式产品时，嵌入式操作系统的选择至关重要。一般而言，在选择嵌入式操作系统时，应考虑以下几个方面。

1. 产品进入市场时间

开发实际产品和一般演示样机不同，产品进入市场时间与操作系统的选择密切相关。例如，选择 Windows CE，能够很快推出产品，进入市场。但伴随的可能是成本高，核心竞争力差。而其他操作系统可能由于编程人员缺乏，或由于技术方面积累不够，影响开发进度。这需要在两者之间抉择。

2. 可移植性

当进行嵌入式产品开发时，软件的可移植性也是要重点考虑的问题之一。良好的软件可移植性可以保证软件方便地在多种平台和系统上运行。

3. 可利用资源

与学术课题研究不同，产品开发以快速、低成本、高质量地推出适合用户需求的产品为目的。集中精力研发出产品的特色，其他功能尽量由操作系统附加或采用第三方产品，因此操作系统的可利用资源对于选型是一个重要参考条件。Linux 和 Windows CE 都有大量的资源可以利用，这是它们被看好的重要原因。有些实时操作系统由于比较封闭，开发时可以利用的资源比较少，因此多数功能需要自己独立开发，从而影响开发进度。

4. 系统定制能力

嵌入式产品不同于传统 PC 的 Wintel (Windows+Intel) 结构的单纯性, 用户的需求是千差万别的, 硬件平台也都不一样, 所以对系统的定制能力提出了要求。Linux 由于其源代码的开放性, 在定制能力方面具有优势。随着 Windows CE 源代码的开放, 以及微软在嵌入式领域力度的加强, 其定制能力也会有所提升。

5. 成本

成本是所有产品开发不得不考虑的问题。选用免费的操作系统, 还是需要支付许可证使用费的商业操作系统, 需要综合权衡以后进行考虑——选择某一系统可能会对其他一系列的因素产生影响, 如对硬件设备的选型、人员投入, 以及公司管理和与其他合作伙伴的共同开发之间的沟通等许多方面的影响。

6. 中文内核支持

国内产品需要对中文的支持。由于操作系统多数是采用西文方式, 因此是否支持双字节编码方式, 是否遵循 GBK, GB18030 等各种国家标准, 是否支持中文输入与处理, 是否提供第三方中文输入接口等, 也是选择操作系统必须考虑的因素。

7. 对嵌入式实时操作系统性能指标的要求

- ① 是否支持多任务功能。目前, 绝大多数嵌入式实时操作系统都支持多任务功能。
- ② 在运行时能创建的任务数是否受限。嵌入式产品的应用程序决定了该上限, 在决定采用一个操作系统前应该估计任务的数量。
- ③ 支持哪些类型的过程间通信机制。要确保在所选的操作系统中具有应用程序选定的过程间通信 (IPC) 机制。
- ④ 时间片是否可调节。时间片如果太小, 相当一部分的处理器时间将消耗在模式切换中, 而如果太大又会影响响应时间。时间片调节功能允许用户通过运行应用程序, 测量其性能, 修改时间片, 再运行这个应用程序的反复过程达到系统最优。
- ⑤ 是否支持优先级继承或某种其他保护措施, 以防止优先级倒置。
- ⑥ 是否支持已经选定的同步对象。
- ⑦ 是否允许选择调度算法。有些操作系统允许从多个调度算法中进行选择, 这样就可试验这些算法, 看哪个最适合应用程序。
- ⑧ 实时操作系统提供的任务切换时间可否接受。一般说来, 不同的操作系统的切换时间差别不大, 而且对大多数的嵌入式系统, 切换时间并不是关键的因素。当系统有大量的任务要求或有严格的时序要求时, 需要比较不同操作系统的时序响应。
- ⑧ 在添加更多任务 (包括线程或过程) 时, 操作系统的性能是否能在可接受的限度内向下调整。
- ⑩ 是否提供优秀的、经过测试的、能处理多任务的编译器和调试器。支持的开发工具, 尤其是调试程序需要与操作系统很好地集成在一起。

8.4.2 嵌入式操作系统的移植

每种嵌入式操作系统都可以支持不同类型和数量的处理器, 从现有的处理器平台到其他处理器平台的移植是嵌入式操作系统应用中经常遇到的问题。下面以 $\mu\text{C}/\text{OS-II}$ 在 TMS320C6711 DSP 上的移植为例, 简单介绍嵌入式操作系统的移植方法和技巧。

在 $\mu\text{C}/\text{OS-II}$ 官方的主页上可以查找到一个比较全面的移植范例列表。但在实际的开发项

目中,如果没有针对项目所采用芯片或开发工具的合适版本,也可以根据需要自己进行移植。

在选定了系统平台和开发工具之后,可按照以下的几个步骤进行 $\mu\text{C}/\text{OS-II}$ 的移植:

1. 深入了解所采用的系统核心

无论项目所采用的系统核心是 MCU, DSP 还是 MPU, 当进行 $\mu\text{C}/\text{OS-II}$ 的移植时, 所需要关注的细节都是相近的。

首先, 是芯片的中断处理机制, 如何开启、屏蔽中断, 可否保存前一次中断状态等。还有, 芯片是否有软中断或是陷阱指令, 又是如何触发的。

此外, 还需关注系统对于存储器的使用机制, 诸如内存的地址空间, 堆栈的增长方向, 有无批量压栈的指令等。

在本例中, 使用的是 TMS320C6711 DSP。这是 TI 公司 6000 系列中的一款浮点型号, 由于其时钟频率非常高, 且采用了超长指令字 (VLIW) 结构、类 RISC 指令集、多级流水等技术, 所以运算性能相当强大, 在通信设备、图像处理、医疗仪器等方面都有着广泛的应用。

在 C6711 中, 中断有 3 种类型, 即复位、不可屏蔽中断 (NMI) 和可屏蔽中断 (INT4-INT15)。可屏蔽中断由 CSR 寄存器控制全局使能, 此外也可用 IER 寄存器分别置位使能。而在 C6711 中并没有软中断机制, 所以 $\mu\text{C}/\text{OS-II}$ 的任务切换需要编写一个专门的函数实现。

此外, C6711 也没有专门的中断返回指令、批量压栈指令, 所以相应的任务切换代码均需编程完成。由于采用了类 RISC 核心, C6711 的内核结构中, 只有 A0-A15 和 B0-B15 这两组 32 位的通用寄存器。

2. 熟悉 C 语言开发工具的特点

无论所使用的系统核心是什么, C 语言开发工具对于 $\mu\text{C}/\text{OS-II}$ 是必不可少的。

最简单的信息可以从开发工具的手册中查找, 比如, C 语言各种数据类型分别编译为多少字节; 是否支持嵌入式汇编, 格式要求怎样; 是否支持 “interrupt” 非标准关键字声明的中断函数; 是否支持汇编代码列表功能, 等等。

这样一些特性, 会给嵌入式开发带来很多便利。TI 的 C 语言开发工具 CCS for C6000 就包含上述的所有功能。

在此基础上, 可以进一步地弄清开发工具的一些技术细节, 以便进行之后真正的移植工作。

首先, 开启 C 编译器的 “汇编代码列表” 功能, 这样编译器就会为每个 C 语言源文件生成其对应的汇编代码文件。

在 CCS 开发环境中设置的方法是, 在菜单 “/Project/Build options” 的 “Feedback” 栏中选择 “Interlisting: Opt/C and ASM(-s)”, 或者也可以直接在 CCS 的 C 编译命令行中加上 “-s” 参数。

然后, 分别编写几个简单的函数进行编译, 比较 C 源代码和编译生成的汇编代码。例如:

```
void FUNC_TEMP(void)
{
    Func_tmp2(); //调用任一个函数
}
```

在 CCS 中编译后生成的 ASM 代码为

```
.asg B15, SP // 宏定义
_FUNC_TEMP:
STW B3,*SP--(8) // 入栈
NOP 2
CALL _Func_tmp2 //-----
MVKL BACK, B3 // 函数调用
```

```

MVKH BACK, B3 //-----
NOP 3
BACK: LDW *++SP(8),B3 // 出栈
NOP 4
RET B3 // 函数返回
NOP 5

```

由此可见，在 CCS 编译器的规则中，B15 寄存器被用做堆栈指针，使用通用存取指令进行栈操作，而且堆栈指针必须以 8 字节为单位改变。

此外，B3 寄存器被用来保存函数调用时的返回地址，在函数执行之前需要入栈保护，直到函数返回前再出栈。

当然，CCS 的 C 编译器对于每个通用寄存器都有约定的用途，但对于 μ C/OS-II 的移植来说，了解以上信息就足够了。

最后，再编写一个用 “interrupt” 关键字声明的函数：

```

interrupt void ISR_TEMP (void)
{
int a;
a=0;
}

```

生成的 ASM 代码为：

```

_ISR_TEMP:
STW B4,*SP--(8) // 入栈
NOP 2
ZERO B4 //-----
STW B4,*+SP(4) // a=0
NOP 2 //-----
B IRP // 中断返回
LDW *++SP(8),B4 // 出栈
NOP 4

```

与前一段代码相比，对于中断函数的编译，有两点不同：

- 函数的返回地址不再使用 B3 寄存器，相应地，也无须将 B3 入栈（IRP 寄存器能自动保存中断发生时的程序地址）。
- 编译器会自动统计中断函数所用到的寄存器，从而在中断一开始将它们全部入栈保护——例如，上述程序段中，只用到了 B4 寄存器。

3. 编写移植代码

在深入了解了系统核心与开发工具的基础上，真正编写移植代码的工作就相对比较简单了。

μ C/OS-II 自身的代码绝大部分都是用 ANSI C 编写的，而且代码的层次结构十分清晰，与平台相关的移植代码仅仅存在于 OS_CPU_A.ASM, OS_CPU_C.C 及 OS_CPU.H 这三个文件中。

在移植的时候，结合前面两个步骤中已经掌握的信息，按照相应的指导来做就可以了。

但是，由于系统核心、开发工具的千差万别，在实际项目中，一般都会有一些处理方法上的不同，这需要特别注意。以 C6711 的移植为例：

(1) 中断的开启和屏蔽

中断的开启和屏蔽的两个宏定义为

```
#define OS_ENTER_CRITICAL() Disable_int()
```

```
#define OS_EXIT_CRITICAL() Enable_int()
```

Disable_int 和 Enable_int 是用汇编语言编写的两个函数。在这里使用了控制状态寄存器 (CSR) 的一个特性——CSR 中除了控制全局中断的 GIE 位之外, 还有一个 PGIE 位, 可用于保存之前的 GIE 状态。

因此, 在 Disable_int 中先将 GIE 的值写入 PGIE, 然后再将 GIE 写 0, 屏蔽中断。而在 Enable_int 中则从 PGIE 读出值, 写入 GIE, 从而回复到之前的中断设置。

这样, 就可以避免使用这两个宏而意外改变了系统的中断状态, 也没有使用堆栈或局部变量。

(2) 任务的切换

前文说过, C6711 中没有软中断机制, 所以任务的切换需要用汇编语言自行编写一个函数 _OSCtxSw 来实现

```
#define OS_TASK_SW() OSCtxSw()
```

在 C6711 中需要入栈保护的寄存器包括 A0-A15, B0-B15, CSR, IER, IRP 和 AMR, 这些再加上当前的程序地址构成一个存储帧, 需要入栈保存。

_OSCtxSw 函数中, 需要像发生了一次中断那样, 将上述存储帧入栈, 然后获取被激活任务的 TCB 指针, 将其存储帧的内容弹出, 从而完成任务切换。

需要特别注意的是, 在这里 OS_TASK_SW 是作为函数调用的, 所以如前文所述, 调用时的当前程序地址是保存在 B3 寄存器中的, 这也就是任务重新激活时的返回地址。

(3) 中断的编写

如前文所述, 如果用 “interrupt” 关键字声明函数, CCS 在编译时, 会自动将该函数中使用到的寄存器进行入栈、出栈保护。

但是, 这会导致各种中断发生时, 出入栈的内容各不相同。这对于 $\mu\text{C}/\text{OS-II}$ 是会引起严重错误的, 因为 $\mu\text{C}/\text{OS-II}$ 要求中断发生时的入栈操作使用与发生任务切换时完全一样的存储帧结构。

因此, 在移植时和基于 $\mu\text{C}/\text{OS-II}$ 进行开发时, 都不应当使用 “interrupt” 关键字, 而应用如下结构编写中断函数:

```
void OSTickISR (void)
{
    DSP_C6x_Save(); // 服务函数, 入栈
    OSIntEnter();
    if (OSIntNesting == 1) // v2.51 版本新增加
    {
        OSTCBCur->OSTCBStkPtr
        =(OS_STK*) DSP_C6x_GetCurrentSP(); // 服务函数
    } // 获取当前 SP 的值
    // 允许中断嵌套, 则在此处开中断
    OSTimeTick();
    OSIntExit();
    DSP_C6x_Resume(); // 服务函数, 出栈
}
```

DSP_C6x_Save 和 DSP_C6x_Resume 是两个服务函数, 分别完成中断的出入栈操作。它们与 OS_TASK_SW 函数的区别在于, 中断发生时的当前程序地址是自动保存在 IRP 寄存器的, 应将其作为任务返回地址, 而不再是 B3。此外, DSP_C6x_Resume 是一个永远不会返回

的函数，在将所有内容出栈后，它就直接跳转回到中断发生前的程序地址处，继续执行。

4. 进行移植的测试

在编写完所有的移植代码之后，就可以编写几个简单的任务程序进行测试。

5. 封装服务函数

最后这个步骤，往往是容易被忽视的，但对于保持项目代码的简洁、易维护有很重要的意义。

μ C/OS-II 的原作者强烈建议将源代码分路径进行存储，例如，例子中的所有源代码就应按如下路径结构存储：

```
\uCOS-II
├──SOURCE // 平台无关代码
│   OS_CORE.C
│   .....
├──TI_C6711 // 系统核心
├──CCS // 开发工具
│   OS_CPU.H
│   OS_CPU_A.ASM
│   OS_CPU_C.C
│
├── DSP_C6x_Service // 服务函数
│   DSP_C6x_Service.H
│   DSP_C6x_Service.ASM
│
└── TEST // 具体的开发项目代码
OS_CFG.H
INCLUDES.H
TEST.C .....
```

如上所示，DSP_C6x_Service 中的服务函数，类似于原作者提供的 80x86 版本中的 PC.C 和 PC.H 文件。在本文的例子中，服务函数则包括了上文提及的中断相关函数，以及系统初始化函数 DSP_C6x_SystemInit()和时钟初始化函数 DSP_C6x_TimerInit()等。

而具体的开发项目代码，则可以分别在“/TI_C6711”路径下新建自己的目录，如同移植测试的“TEST”项目，而无须再关注 μ C/OS-II 的源代码和服务函数。这样，就可以避免不必要的编译错误，也便于开发项目的维护。

8.4.3 嵌入式软件的发展趋势

对所有嵌入式系统来说，硬件是支撑，软件是灵魂，它可以提供灵活多样的功能。由于嵌入式系统应用广泛，嵌入式软件在整个软件产业中占据了重要地位，并受到世界各国的广泛关注。如今已成为信息产业中重要的发展方向之一。

新一轮汽车、通信、信息电器、医疗、军事等行业的巨大的智能化装备需求，拉动了嵌入式软件及系统的发展。与传统的通用计算机系统不同，嵌入式系统面向特定应用领域，根据应用需求定制开发，并随着智能化产品的普遍需求渗透到各行各业。随着硬件技术的不断革新，硬件平台的处理能力不断增强，硬件成本不断下降，嵌入式软件已成为产品的数字化改造、智能化增值的关键性、带动性技术。

SoC 技术是微电子技术发展的一个新的里程碑，并已成为当今超大规模集成电咱（IC）的发展趋势，为 IC 产业提供了前所未有的广阔市场和难得的发展机遇。迅猛发展的 SoC 技术推进了

嵌入式软件为灵魂与核心的软件与硬件系统进一步融合嵌入，改变了传统嵌入式系统的设计观念。基于 IP 构件库的设计技术将成为嵌入式系统设计的主流。IP 构件库技术正在造就一个新兴的软件行业。

随着因特网技术的不断发展和大规模应用，大量的嵌入式设备急需网络连接来提升其服务能力和应用价值。同时，以人为中心的普适计算技术正推动新一轮的信息技术的革命，嵌入式设备将以各种形态分布在人类的生存环境中，提供更加人性化、自然化的服务。互联网的“深度”联网和普适计算“纵向”普及所带来的计算挑战，催生了新型嵌入式软件技术，将推动嵌入式软件技术向“纵深”发展。

硬件技术的进步，微处理器从 8 位到 16 位、32 位甚至 64 位，推动了嵌入式系统软件向运行速度更快，支持功能更强，应用开发更便捷的方向不断发展。近十年来，嵌入式操作系统得到飞速的发展。从支持单一品种的处理到支持多品种的，从单一内核到多内核并提供其他功能模块，如文件系统、TCP/IP 网络系统、窗口图形系统等，形成了包括嵌入式操作系统、中间平台软件在内的嵌入式软件体系。

嵌入式系统应用不断深入，产业化程度不断提升。新的应用环境和产业化需求对嵌入式系统软件提出了更加严格的要求。嵌入式操作系统内核不仅需要具有微型化、高实时性等基本特征，还将向高可靠性、自适应性、构件组件化方向发展。支撑开发环境将更加集成化、自动化、人性化。系统软件对无线通信和能源管理的功能支持将日益重要。

行业性开放系统正日趋流行。统一的行业标准具有开放、设计技术共享、软硬件并重、构件兼容、维护方便和合作生产的特点，是增强行业性产品竞争能力的有效手段。嵌入式操作系统本身正走向开放化、标准化。Linux 正逐渐进入嵌入式操作系统的主流。J2ME 技术也对嵌入式软件的发展产生深远影响。目前自由软件技术备受青睐，对软件技术的发展产生了巨大的推动作用，为嵌入式软件技术的发展提供了良好的机遇。

习题与思考题

- 1 何为嵌入式系统？简述嵌入式系统的体系结构。
- 2 简述嵌入式操作系统的概念及其分类。
- 3 简述嵌入式操作系统的主要特点。使用嵌入式操作系统有哪些优缺点？
- 4 常用的嵌入式操作系统的任务调度算法有哪几种？简述其各自的特点。
- 5 简述嵌入式 Windows 的主要产品及其特点。
- 6 试比较分析两种常见的嵌入式操作系统 $\mu\text{C}/\text{OS-II}$ 和 μCLinux ？
- 7 嵌入式操作系统的实时性能通常用哪些指标进行衡量？
- 8 分析一种你比较熟悉的嵌入式操作系统，简述其主要性能指标、应用领域与优缺点。
- 9 选择适合的嵌入式操作系统一般需考虑哪些问题？
- 10 观察一下你身边有哪些产品应用了嵌入式操作系统？

第9章 网络与现场总线

网络与通信技术的飞速发展，使世界变成了地球村，深刻地改变了现代人的生活方式、表达方式和思维方式，也引起了控制系统结构的变革。以网络集成自动化系统为基础的企业信息系统已成为当今自动化领域技术发展的热点之一，信息的交换使管理、控制的各个层次直至工厂的现场设备层都可以沟通，可以覆盖所有工段、车间、工厂、企业乃至世界各地。

9.1 数字通信与网络

计算机网络中，多台计算机以总线为纽带互连，完成数字通信任务；控制领域的微机局域网网络也是这样构成的。

9.1.1 总线的基本概念

总线是遵循同一技术规范的连接与操作方式来传输信号或信息的公共路径。一组设备通过总线连在一起称为“总线段”（Bus Segment）。多个总线段相互连接就成为一个网络系统。

使用总线的规则称为“总线协议”（Bus Protocol）。这是一套事先规定的、必须共同遵守的规约。

可以在总线上主动发起信息传输的设备称为“总线主设备”（Bus Master），又称为“命令者”；只能挂接在总线上，对总线信息进行接收、查询的设备称为“总线从设备”（Bus Slaver）。被总线主设备连上的从设备称为“响应者”（Responder）。

一次总线操作包括命令者与响应者之间的连接 → 数据传送 → 脱开。某一命令者与一个或多个响应者连接上以后，就可以开始数据的读写操作。

命令者与一个或多个从设备建立起联系的总线操作称为寻址，通常寻址方式有以下三种：

- 物理寻址 用于选择某一总线段上某一特定位置的从设备作为响应者。大多数从设备都包含有多个寄存器，因而物理寻址常常有辅助寻址，以选择响应者的特定寄存器或某一功能。
- 逻辑寻址 用于指定存储单元的某一个通用区，而并不顾及这些存储单元在设备中的物理分布。某一设备监测到总线上的地址信号，看其是否与分配给它的逻辑地址相符，如果相符，它就成为响应者。物理寻址与逻辑寻址的区别在于，前者选择的设备与物理位置有关，而后者选择的设备与物理位置无关。
- 广播寻址 用于选择多个响应者。命令者把地址信息放在总线上，地址相符的从设备就“连上”（Connect）。能使多个从设备连上的地址称为“广播地址”（Broadcast Addresses）。为了确保所选的全部从设备都能响应，系统需要有适应这种操作的定时机制。

系统总线一般采用逻辑寻址，一些新的系统总线还常常兼备上述其他寻址方式，现场总线则较多采用物理寻址和广播寻址。

任何时刻都只允许一个主设备占有总线。总线上如果有多个主设备，在传送信息的过程中，有可能发生“冲突”（Contention）。为决定哪一个主设备占有总线，需要进行总线占

有权的“仲裁”(Arbitration)。总线仲裁操作和数据传送操作完全分开且并行工作,使得总线占有权的交接过程不会耽误总线操作。仲裁的方案通常有:

- 集中仲裁 操作由一个仲裁单元完成。若两个以上主设备同时请求使用总线时,仲裁单元根据优先级方案进行仲裁:有的方案允许高优先级的主设备一直占有总线,而无限期地否决低优先级主设备的请求;另一些方案则不允许某一主设备“霸占”总线。
- 分散仲裁 操作在各个主设备中完成。当主设备将自己的优先级代码置放在总线上时,一个仲裁周期开始。仲裁周期结束时,只有最高优先级代码仍置放在总线上,优先级与此相符的主设备在下一时刻占有总线。

总线操作用“定时”(Timing)信号进行同步,定时信号用于指明总线上的数据和地址在什么时刻是有效的。定时信号有异步和同步两种。

在大多数同步总线系统中,定时时钟信号由系统统一提供。总线状态的改变只出现在时钟的固定时刻。通常根据连在总线上响应最慢的设备设置时钟的速率来确定总线周期的持续时间,为了避免因与低速设备通信而降低系统的整体性能,在总线标准中规定允许插入等待周期。例如,某一慢速设备为完成所请求的操作,可设置等待信号,直至该操作完成;等待信号撤销以后,系统恢复正常的同步操作。使用等待约定的同步系统和异步系统均有总线超时处理,在规定的时间内没有得到响应者的响应,系统就夭折该总线周期。

在异步总线系统中,命令者发出选通定时信号,表明总线上的信息有效,响应者回送一个应答定时信号,命令者收到该应答信号后,证实响应者确实进行了响应。这一过程称为“握手”(Handshake)。

在总线上传送信息时,会因噪声和串扰而出错,高性能的总线一般都设置检错和容错机制,以实现传送地址和数据过程的出错检测,并将故障单元隔离或更换。

9.1.2 数字信号传输与通信系统

1. 数字通信系统的组成

通信系统是传递信息所需的一切技术设备的总和,包括信息源、发送设备、传输介质、信息接收者和接收设备。数字通信系统传输的数据是数字化了的信息。单向数字通信系统的结构如图9.1所示。

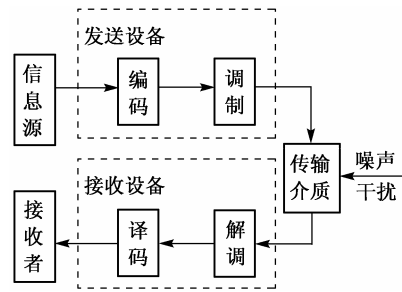


图 9.1 数字通信系统的组成

信息源中,模拟信息源(如模拟式电话机、电视摄像机)输出幅度连续变化的信号,离散信息源(如计算机)输出离散的符号序列或文字。通过采样和量化,可以将模拟信息变换为离散信息。

发送设备的基本功能是为不同种类和速率的信息源与传输媒介相匹配,通常是将信息源产生的信息经过编码,并变换为便于传送的信号形式,送往传输介质。

编码包括信源编码与信道编码两部分。信源编码把连续消息变换为数字信号,信道编码则使数字信号与传输介质匹配,提高传输的可靠性或有效性。调制是多种变换方式中最常见的一种。

发送设备还包括为达到某些特殊要求所进行的各种处理,如多路复用、保密处理、纠错编码处理等。

传输介质是发送设备到接收设备之间信号传递所经过的媒介,例如,电磁波、红外线等

无线传输介质，各种电缆、光缆、双绞线等有线传输介质。传输过程中，必然会引入热噪声、衰减、脉冲等干扰。介质的固有特性和干扰特性直接关系到变换方式的选取。

接收设备的基本功能是完成发送设备的逆变换（解调、译码、解密等），从带有干扰的信号中正确恢复出原始信息。对于多路复用信号，还包括解除多路复用，实现正确分路（或称为输出扫描）。

双向通信要求通信双方都有发送设备和接收设备，如果两个方向共用一个传输媒介，则必须采用分频或分时的办法。信息的传输系统和交换系统组成完整的通信系统，直至构成复杂的通信网络。

2. 数据编码

码元是传输数据的基本单位。在计算机网络通信中所传输的大多数为二源码，每一位只能在1或0两个状态中取一个。

数据编码是指通信系统中以何种物理信号的形式来表达数据0, 1状态。若分别用模拟信号的不同幅度、不同频率、不同相位来表达，则称为模拟数据编码。若用高低电平的矩形脉冲信号来表达，则称为数字数据编码。

采用数字数据编码，在基本不改变数据信号频率的情况下，直接传输数据信号的传输方式，称为基带传输。基带传输可以达到较高的数据传输速率，是数据通信的基本方式。

编码有多种方式，例如：

- 信号电平是单极性的单极性码和信号电平为正、负两种极性的双极性码；
- 每一位二进制信息传输后，均返回零电平的归零码（RZ）和在整个码元时间内维持有效电平的非归零码（NRZ）；
- 用电平变化代表逻辑“1”，不变化代表“0”的差分码。

根据信息传输方式，还可分为平衡传输和非平衡传输。平衡传输指无论“0”或“1”都是传输格式的一部分；而在非平衡传输中，只有“1”被传输，“0”则以在指定的时刻没有脉冲来表示。

实际的基带传输方式往往是上述几种方式的结合。例如，图 9.2 所示的曼彻斯特编码(Manchester Encoding)就属于平衡、归零、单极性方式，在每个比特期间的中间点，总有一次信号电平的跳变，因此携带有信号传送的同步信息，而不需要另外传送同步信号。

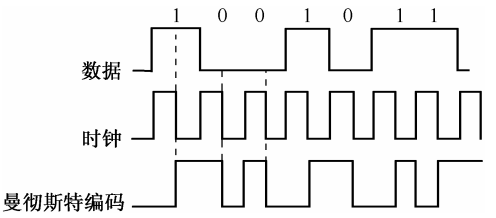


图 9.2 曼彻斯特编码过程与波形

理想的方波信号包含从零到无限高的频率成分，但传输线中不可避免地存在着分布电容，允许传输的带宽是有限的，所以不可能要求方波波形传输完全不失真。为了与线路传输特性匹配，除很近距离传输外，一般可用低通滤波器将矩形波整形成变换点比较圆滑的基带信号，由接收端在每个码元的最大值（中心点）取样复原。

3. 通信系统的性能

通信系统最重要的性能是信息传输的有效性和可靠性。

信息速率是衡量数字通信系统有效性的指标。二进制信号的信息速率用比特每秒（bps）作为单位，常称为比特率。如比特率 9600 bps，表示每秒可传输 9600 个二进制脉冲。常用的标准数据信号的信息速率为 50 bps，100 bps，200 bps，300 bps，600 bps，1200 bps，2400 bps，4800 bps，9600 bps、240 kbps，1 Mbps，10 Mbps，1000 Mbps 等。

信道的最大数据传输速率受信道带宽的限制。当信道一定时,信息速率越高,有效性越好。

误码率是码元在数据传输系统中被传错的概率,是衡量数字通信系统可靠性的指标。当传输的码元总数 N 足够大时,误码率数值上近似为被传输错的码元数 N_e 与 N 之比。

根据测试,目前电话线路在 4800~9600 bps 传输速率时平均误码率在 $10^{-2} \sim 10^{-4}$ 之间。为了满足计算机通信的平均误码率低于 10^{-9} 的要求,必须采取差错控制技术才能利用普通通信信道。

误码率应根据实际传输要求提出。在数据传输速率确定后,误码率要求越低,则数据传输系统设备越复杂,造价越高。

4. 信号的传输方式

(1) 基带传输

基带传输是最基本的数据传输方式,即按数据波的原样,不包含任何调制,在数字通信的信道上直接传送数据。基带传输不适于传输语言、图像等信息。

目前,大部分微机局域网包括控制局域网,都是采用基带传输方式的基带网。基带网的特点是,信号按位流形式传输,整个系统不用调制解调器,降低了价格。传输介质较宽带网便宜,可以达到较高的数据传输速率,但其传输距离一般不超过 25 km,传输距离越长,质量越低。基带网中,线路工作方式只能为半双工方式或单工方式。

(2) 载波传输

载波传输采用数字信号对载波进行调制后进行传输。最基本的调制方式有幅度键控(ASK)、频移键控(FSK)、相移键控(PSK)三种。

(3) 宽带网

宽带网的数据传输速率可以比基带网高得多,还可划分为多条基带信道,为数据、文字、语音、图像等多种信号提供良好的通信路径。

(4) 异步传输模式(Asynchronous Transfer Mode, ATM)

ATM 支持多媒体通信,包括数据、语音和视频信号等数字化信息的传输与交换,按需分配带宽,具有低延迟特性,速率可达数 Gbps,是一种将时分交换与统计复用融为一体的、面向连接且分组长度固定的高速传输模式。ATM 首先将信息切割成块,并在块前加上信头(包括地址、优先级等控制信息),构成信元(Cell)。固定长度的信元定时出现,融合了线路传送模式和分组传送模式的优点,因此可采用硬件高速地对信头进行识别和交换处理。

对于实现超级信息高速公路来说,ATM 有着广阔的发展前景。

5. 多路转换

在一条物理通道上同时传送多路信息的技术称为多路转换(Multiplexing)。常用的为频分多路转换和时分多路转换:

- 时分多路转换(Time Division Multiplexing, TDM) 将线路用于传输的时间划分成若干个时间片,每个用户分得一个时间片,这些时间片是预先分配好的,而且固定不变。时分多路转换不仅用于传输数字信号,也可用于传输模拟信号。
- 频分多路转换(Frequency Division Multiplexing, FDM) 将一条具有一定带宽的线路划分成若干条占有较小带宽的信道,各条信道中心频率不重合,每个信道之间相距一定的频率间隔,每个用户使用一条频道。

9.1.3 计算机局域网

微型计算机和以计算机为核心的各种设备的广泛使用,不但要求分散而有效的数据处理

与计算能力，还要求能彼此沟通信息，共享资源，协同工作。计算机网络就是用通信线路（包括通信卫星、微波通信等技术）将各计算机连接起来，以实现资源共享和作业分布处理的计算机群。

计算机网络按地域范围可分为远程网和局域网。远程网的跨越范围可从几十千米到几万千米，传输线造价很高。

局域网的距离只限于几十米到 25 km，一般为 10 km 以内，其传输速率较高，在 0.1 Mbps 到 10 Gbps 之间。其误码率也很低，为 10^{-8} 至 10^{-11} 。

网络拓扑结构、信号方式、访问控制方式和传输介质是影响网络性能的主要因素。

1. 网络拓扑结构

网络的拓扑结构是指网络中节点的互连形式。图9.3所示的星形、环形、总线形和树形是较为常见网络拓扑结构。

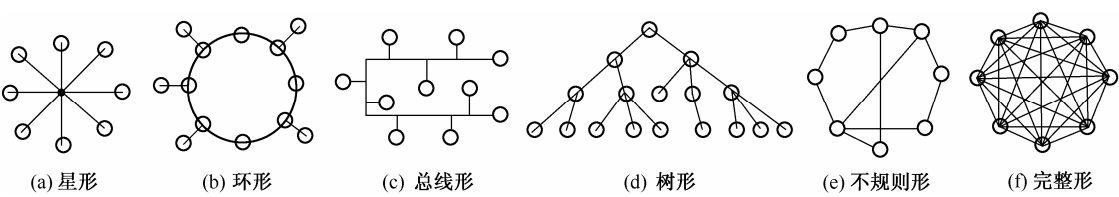


图 9.3 网络拓扑结构

(1) 星形拓扑

星形拓扑采用集中式通信控制策略，所有通信均由中央节点控制，任何两站之间通信都通过中央节点进行。中央节点必须建立和维持许多并行数据通路，需要有较复杂的网络处理功能；而每个站的结构很简单，通信处理负担也很小，只需满足点-点链路简单的通信要求。

星形拓扑采用电路交换，可实现数据通信量的综合，适用于低数据率设备和终端密集的要求。

(2) 环形拓扑

环形拓扑网络中，中继器进行点-点链路连接，构成一个封闭的环路。中继器接收前驱站发来的数据后，按原来速度一位一位地从另一条链路发送出去，数据沿顺时针或逆时针方向在网上环行。每个工作站通过中继器再连至网络，数据拆成分组加上控制信息插入环上，通过其他中继器到达目的站。哪个站何时能向环上插入分组由访问控制确定。

环形拓扑的网络设备只是很简单中继器，而工作站则需提供拆包和存取控制逻辑等较复杂的功能。中继器之间可使用高速链路（如光纤），可提供更大的吞吐量，适用于工业环境。

(3) 总线拓扑

总线拓扑的传输介质是一条总线，工作站通过相应硬件接口接至总线上。一个站发送数据，所有其他站都能接收，称为多点式或广播式。因为所有节点共享一条传输链路，一次只允许一个站发信息，需有某种存取控制方式，确定下一个可以发送的站。信息也是按分组发送，达到目的站后，经过地址识别，将信息复制下来。

(4) 树形拓扑

树形拓扑是总线拓扑的扩展形式，传输介质是不封闭的分支电缆。它和总线拓扑一样，一个站发送数据，其他站都能接收。树形拓扑的适应性很强，可适用于很宽的范围，如对网络设备的数量、数据率和数据类型等没有太多限制，可达到很高的带宽。树形结构在单个局

域网系统中采用不多。如果把多个总线形或星形网连在一起，或连到另一个大型机或一个环形网上，就形成了树形拓扑结构，适合于分主次、分等级的层次型管理系统。

2. 网络的传输介质

传输介质是网络中连接收发双方的物理通路，也是通信中实际传送信息的载体。网络中常用的传输介质有电话线、同轴电缆、双绞线、光纤、光缆、无线与卫星通信。传输介质的特性对网络中数据通信质量影响很大，这些特性主要是：

- 物理特性：传输介质物理结构的描述；
- 传输特性：传输介质允许传送数字或模拟信号，以及调制技术、传输容量、传输的频率范围；
- 连通特性：允许点-点或多点连接；
- 地理范围：传输介质最大传输距离；
- 抗干扰性：传输介质防止噪声与电磁干扰对传输数据影响的能力；
- 相对价格：器件、安装与维护费用。

选择传输介质时，要考虑到网络拓扑、网络连接方式的限制、网络通信量、系统可靠性的要求、所要传输的数据类型、网络覆盖的地理范围、节点间的距离等因素。

(1) 双绞线

无论对于模拟信号还是数字信号，双绞线都是最通用的传输介质。

双绞线由按规则螺旋结构排列的两根或四根绝缘线组成，螺旋排列使各线对之间的电磁干扰最小。一对线可以作为一条通信线路。

双绞线最普遍的应用是语音信号的模拟传输，电话线路就是一种双绞线。在一条双绞线上，使用频分多路复用技术可以进行多个音频通道的多路复用。如每个通道占用 4 kHz 带宽，并在相邻通道之间保留适当的隔离频带，双绞线使用的带宽可达 268 kHz，可以复用 24 条音频通道的传输。

当使用双绞线或调制解调器传输模拟数据信号时，数据传输速率可达 9600 bps，24 条音频通道总的数据传输速率可达 230 kbps。

双绞线可以用于点-点连接，也可用于多点连接。当用做远程中继线时，最大距离可达 15 km；当用于 10 Mbps 局域网时，与集线器的最大距离为 100 m。

双绞线的抗干扰性取决于一束线中相邻线对的扭曲长度及适当的屏蔽。低频传输时，抗干扰能力与同轴电缆相当，在 10~100 kHz 时，其抗干扰能力低于同轴电缆。

双绞线的价格低于其他传输介质，并且安装、维护方便。

(2) 同轴电缆

同轴电缆在网络中应用十分广泛，可以按通频带分为基带和宽带两类。基带同轴电缆一般仅用于数字数据信号传输。宽带同轴电缆既可以只用于一条通信信道的高速数字通信（单通道宽带），也可以使用频分多路复用方法和各种调制方案，将一条宽带同轴电缆的频带划分成多条通信信道。

同轴电缆由内导体、绝缘层、外导体及外部保护层组成，如图9.4所示。同轴介质的特性参数由内、外导体及绝缘层的电参数和机械尺寸决定。

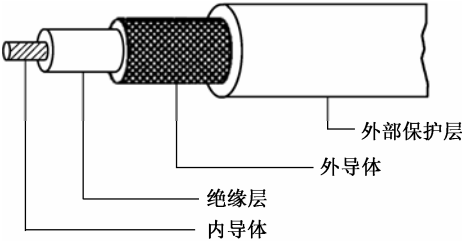


图 9.4 同轴电缆结构示意图

同轴电缆的特征阻抗与内、外导体的几何尺寸、绝缘层介质常数有关。网络中常用的同轴电缆有：

- Ethernet 使用的同轴电缆，特征阻抗为 $50\ \Omega$ ，用于基带传输，传输速率可达 10 Mbps 以上。
- 公用有线电视使用的 CATV 电缆，特征阻抗为 $75\ \Omega$ ，既可以用于模拟信号传输，也可用于数字信号传输。当用于模拟信号传输时，CATV 电缆带宽可达 400 MHz。可以用频分多路复用 FDM 技术，将 CATV 电缆的带宽分成多个传输模拟信号或数字信号的通道。

同轴电缆支持点点连接，也支持多点连接。宽带同轴电缆可支持数千台设备的连接，最大距离可达几十千米。基带同轴电缆可支持数百台设备的连接，最大距离限制为几千米。

同轴电缆的抗干扰能力较强，造价介于双绞线与光缆之间，维护方便。

(3) 光缆

光导纤维是一种直径为 $50\sim100\ \mu\text{m}$ 的柔软、能传导光波的介质，常用材料是各种玻璃和塑料；光的传输距离与光纤的光损耗有关，用新型氟玻璃制成的光纤，可以把光信号传输到太平洋彼岸而不需任何中继站。

在实际使用时，常把千百根光纤组合在一起并加以增强处理，制成像电缆一样的光缆，这样既提高了光纤的强度，又大大增加了通信容量。光缆具有低损耗、宽频带、高数据传输速率、低误码率、抗干扰性好等特点，价格下降也比较快，是一种应用前途广泛的网络传输介质。

光纤的结构如图 9-5(a)所示，经过编码的光信号传输过程如图 9-5(b)所示。由于光纤的折射系数高于外部包层的折射系数，形成光波在光纤与包层界面上的全反射，以小角度进入的光波沿光纤按全反射方式向前传播。

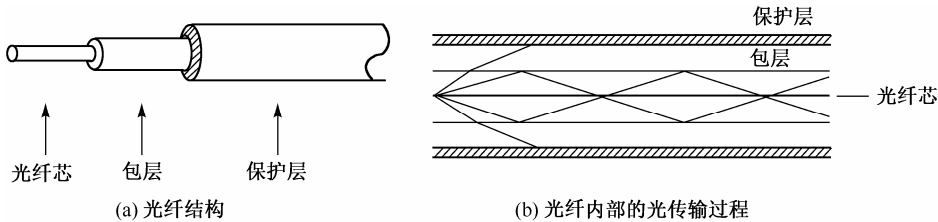


图 9.5 光纤结构与传输

光纤可以看做传输频率为 $10^{14}\sim10^{15}\ \text{Hz}$ 光波的导线，这一范围覆盖了可见光谱与部分红外光谱。典型的光纤传输系统的结构如图 9.6 所示。光纤发送端采用两种光源：发光二极管 LED 和注入型激光二极管 ILD。接收端将光信号转换成电信号时使用光电二极管 PIN 检波器或 APD 检波器。光载波调制方法采用振幅键控 ASK 调制方法，即亮度调制。光纤传输速率可达几千 Mbps 以上。

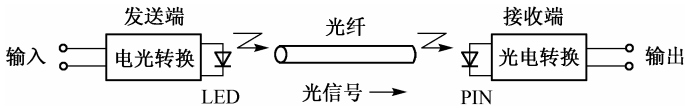


图 9.6 光纤传输系统示意图

光纤传输分为单模与多模两类。单模光纤（Single-Mode Fiber, SMF）是指光纤的光信号仅与光纤轴成单个可分辨角度的单光线传输，而多模光纤（Multi-Mode Fiber, MMF）的光信号与光纤轴成多个可分辨角度的多光线传输。单模光纤性能优于多模光纤。

光纤最普遍的连接方法是点-点方式。

光纤信号衰减极小，一般在 6~8 km 距离内可以不使用中继器，实现高速率数据传输。光纤传输的安全性与保密性极好，并且不受外界电磁干扰与噪声的影响，能在长距离、高速率传输中保持低误码率。双绞线典型的误码率在 $10^{-5} \sim 10^{-6}$ 之间，基带同轴电缆为 10^{-7} ，宽带同轴电缆为 10^{-9} ，而光纤误码率可以低于 10^{-10} 。

(4) 无线数据通信

无线数据通信适应了对不受限制，无论在何时、何地、与任何人都能够进行任何内容（包括数据、语音、图像等）的通信需求。

① 无线广域网（Wireless WAN, WWAN）。无线广域网一般是指对公众运营的移动数据网络，例如，通用分组无线业务（GPRS），可以给用户提供移动环境下的高速语音和数据传输。

② 无线局域网（Wireless LAN, WLAN）。用无线电波、激光、红外线等来代替部分或全部线路，成为网络设备之间主要的通信传输介质，从而构成了无线局域网。

IEEE802.11 标准中，物理层定义了数据传输的信号特征和调制方法，其技术规范包括红外线技术规范、跳频扩频（FHSS）和直接序列扩频（DSSS）三种射频技术规范。

- 红外线局域网：可以分为室内和室外通信两种类型。室外红外无线通信主要指空间中的红外无线通信，即激光通信，而应用更广泛、更具实用价值的是室内红外无线数据通信。使用小于 $1\mu\text{m}$ 微米波长的红外线作为传输媒体，调制方式为 16 PPM（脉冲位置调制），具有较强的方向性，受光线的干扰较大，一般仅用于直线视距数据传输。
- 跳频扩频局域网：载波频率在很宽的频带范围内按某种图案（伪随机序列 MAC 层控制）进行跳变，接收端首先从发送来的跳频信号中分离出跳频同步信号，使接收机的伪随机序列控制的频率跳变与接收到的跳频信号同步（即发送者和接收者在传输过程中同步的转移频率），输出的被同步后的载波经解调后，获得发送端送来的信息。跳变频率至少为 1 跳/秒。跳变过程中将能量扩散到整个频带内，利用它实现扩频并最终获得传输增益。
- 直接序列扩频局域网：这种通信方式在传输过程中以宽带方式通信，而数据发送和接收端都以窄带方式进行。数据发送端将要发送的信号用伪随机码（PN 码）扩展到一个比原始信号频带宽得多的频带上去，以实现扩频。在接收端，利用与发送端相同的 PN 码进行相关解扩，恢复出发送信号。由于干扰信号与 PN 码不相关，解扩后落入信号通带内的干扰信号功率降低到原来的 $1/G$ （ G 为扩频增益系数），从而提高了相关器的输出信噪比，达到抗干扰目的。

DSSS 和 FHSS 无线局域网都使用覆盖范围较广的无线电波作为媒体，具有抗干扰、抗噪声、抗衰减和保密性能好等优点。

③ 无线个人区域网络（Wireless Personal Area Network, WPAN）。无线个人区域网络最为现实的应用是在家庭或办公室的各种设备、电话系统及有线网络间交换数据。主要有视距红外传输，如广泛应用于计算机和无线通信设备的 IrDA 规范；短距离无线电链路，如作为分散设备无线互联协议的蓝牙（Bluetooth）规范。

3. 介质访问控制方式

如前所述，在总线和环形拓扑中，网上设备必须共享传输线路。为解决在同一时间有几个设备争用传输介质的问题，必须有控制方式来协调各设备访问介质的顺序。

通信中对介质的访问可以是随机的，常用的控制方式为 CSMA/CD；也可以是受控的，常用的控制方式为令牌总线、令牌环。

(1) CSMA/CD（载波监听多路访问/冲突检测）

CSMA 控制方式对任何工作站都没有预约发送时间，即各工作站可在任何时刻，任意地点访问介质。随机发送的工作站必须在网络上争用传输介质，故称为争用总线技术。如果同一时刻有多个工作站向传输线路发送信息，这些信息会在传输线上相互混淆而遭破坏，称为“冲突”。为尽量避免由于竞争引起的冲突，每个工作站在发送信息之前，都要监听传输线上是否有信息在发送，这就是“载波监听”。

CSMA 的控制方案是“先听再讲”。一个站要发送，首先需监听总线，以决定介质上是否存在其他站的发送信号。如果介质空闲，则可以发送；如果介质忙，即存在着其他站的发送信号，则等待一定间隔后重试。监听总线状态后的退避算法有以下三种：

- 不坚持 CSMA。介质空闲，则发送；介质忙，则等待一段随机时间后，再重复上述操作。这种算法利用随机的重传时间来降低冲突的概率，缺点是介质的利用率较低，即使有几个站要发送数据，介质仍然可能处于空闲状态而让几个站都等待。
- 1—坚持 CSMA。介质空闲，则发送；介质忙，继续监听，直到介质空闲，立即发送。假如冲突发生，则等待一段随机时间后，再重复上述操作。这种算法提高了介质利用率，缺点是假如有两个或两个以上的站点要发送，冲突就不可避免。
- P—坚持 CSMA。介质空闲，则以 P 的概率发送，或以 $(1-P)$ 的概率延迟一个单位时间后重复处理，该时间单位等于最大的传输延迟；介质忙，继续监听直到介质空闲，重复上述操作。这是种折中的算法，试图降低像 1—坚持 CSMA 算法中的冲突概率，另一方面又减少像不坚持 CSMA 算法中的介质浪费。

假如介质忙时，有 N 个站有数据等待发送，则当前的发送完成时，有 NP 个站企图发送，如果 P 值选得过大，使 $NP > 1$ ，则冲突不可避免。最坏的情况是，随着冲突概率的不断增大，吞吐率会降为 0。所以，必须选择 P 值使 $NP < 1$ ；如果 P 值选得过小，则通道利用率又会大大降低。

由于传输线上不可避免地存在传输延迟，因此有可能多个站同时监听到线上空闲并开始发送，从而导致冲突。故每个工作站发送信息之后，还要继续监听线路，判定是否有其他站正与本站同时向传输线发送，一旦发现，便中止当前发送，这就是“冲突检测”。

CSMA/CD 协议已广泛应用于局域网。每个站在发送帧期间，同时有检测冲突的能力，即所谓“边讲边听”。一旦检测到冲突，就立即停止发送，并向总线上发一串阻塞信号，通知总线上各站冲突已发生，这样不至于因白白传送已损坏的帧而浪费通道的容量。

基带总线所需的冲突检测时间等于任意两个站之间最大传输延迟的 2 倍。

宽带总线所需的冲突检测时间等于任意两个站之间最大传输延迟的 4 倍。

(2) 令牌（标记）访问控制方式

为了解决各站点随机发报带来的冲突问题，可采用控制各站访问介质顺序和时间的发报方式。令牌方式是一种按一定顺序在各站点传递令牌的方法。拥有令牌的站点才有发报权，其他站点仅允许接收报文。站点在发送完毕后，便将令牌交给网上下一个站点，如果该站点没有报文需要发送，便把令牌顺次传给下一个站点。令牌访问原理可用于环形网络，构成令牌环形网；也可用于总线网，构成令牌总线网络。

① 令牌环 (Token-Ring) 方式

表示发送权的令牌在环形信道上不断循环, 环上每个相应站点都可获得发报权, 而任何时刻只会有一个站点利用环路传送报文, 因而在环路上保证不会发生访问冲突。

令牌传递方式工作原理如图9.7(a)所示。图中各站点的一个入口和一个出口分别与环形信道相连。通信处理器中有缓冲器, 用来存储转发信息。

在网上传输的帧的大致格式如图9.7(b)所示。开始标志表示帧的开始, 目的地址是该帧的接收站地址, 源地址是该帧的发送站地址, 报文即为帧中的数据, 检验和用于检查帧传输中的差错情况, 状态用于指示此帧发出后是否为目的站所接收, 结束标志用于表示此帧的结束。



图 9.7 环形网示意图

若 A 站要发送数据给 C 站, 则 A 站把目的地址和要发送的数据交给本站的通信处理器并组织成帧。一旦本站从环上得到令牌, 就从出口发出该帧。B 从其入口收到此帧后, 查看目的地址与本站地址不符, 便将原帧依次转发给 C。C 在查看目的地址时, 得知此帧是给本站的, 便根据检验和进行查错, 如果传输的帧无错误, 便将帧中的数据收下, 并修改状态位, 表示此帧已被正确接收, 这时 C 再把修改了状态位的原帧沿 D, E 送回 A。A 从返回的帧状态位得知发送成功, 从环上取消此帧, 再把令牌转交给 B, 这样就完成了一次站间通信过程。

采用令牌环方式的局域网, 网上每一个站点都知道信息的来去动向, 并能估算出信息传输的延迟时间, 信息传输的确定性较高。

② 令牌传递总线方式

令牌传递总线方式已列为 IEEE 802.4 协议标准。

如图9.8所示, 从物理上看, 这是一种总线网络拓扑结构的局域网, 站点共享的传输介质为总线。但从逻辑上看, 这是一种环形结构的局域网, 接在总线上的站组成一个逻辑环, 每个站被赋予一个顺序的逻辑位置, 每站都了解先行站 (PS) 和后继站 (NS) 的地址。总线上

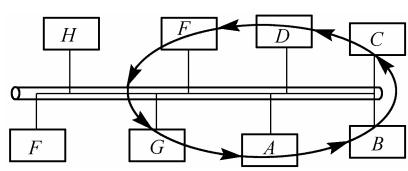


图 9.8 令牌传递总线访问控制

各站的物理位置与逻辑位置无关。

和令牌环方式一样, 令牌传递总线方式的传输过程也由交替进行的数据传输阶段和令牌传送阶段组成。网上站 (如图9.8中的 H, F) 也可以退出环成为非活动站点, 仅能响应询问或请求应答。

正常运行时, 完成了发送的站点将令牌送给下一个站, 从逻辑上看, 令牌是按地址的递减顺序传送至下一个站点; 但从物理上看, 是带有目的地址的令牌帧广播到总线上所有的站点, 由识别出地址相符的目的站接收。取得令牌的站如果有报文要发送, 则发送报文后将令牌送至下一个站; 如果没有报文要发送, 则立即把令牌送至下一个站。站点接收到令牌的过程是顺序依次进行的, 所有站点都有公平的访问权。

假设各站都没有报文发送, 令牌的形式为 01111111, 称为空标记; 要发送报文的站需等

待空标记到来，然后将它改为忙标记，即 01111110；紧跟着忙标记，该站把数据帧发送到环上。由于标记是忙状态，其他站不能发送，只能等待。等待访问时间是微机控制局域网的关键参数，可以根据需求，选定网中的站点数及最大的报文长度，从而保证在限定的时间内，每一站点都有取得令牌的机会。

令牌传递总线访问控制还提供了不同的优先级。

令牌传递总线介质访问控制应具备以下功能：

- 令牌传递算法：逻辑环按递减的站地址次序组成，刚发完帧的站将令牌传给后继站，后继站应立即发送数据或令牌帧，原先释放令牌的站可以根据监听到的总线上的信号，确认后继站获得了令牌。
- 逻辑环的初始化：网络开始启动时或在运行中，所有站点活动的时间由于某种原因超过规定，都需要进行逻辑环的初始化，即生成一个顺序访问的次序。初始化的过程是一个争用的过程，争用的结果只有一个站能获得标记，其他的站可以以后插入。
- 站插入：逻辑环上的每个站应周期性地使新的活动站点有机会插入环中。当同时有几个站要插入时，可以采用带有响应窗口的争用处理算法。
- 退出环路：一个不活动的工作站可以从环路中退出，并将其先行站和后继站连接起来。
- 故障恢复：网络中可能出现差错，例如，令牌丢失或产生多个令牌，应能识别处理，恢复正常工作。

4. 数据交换

计算机网络中，通信的基本交换方式分为两类：线路交换与存储转发交换。存储转发交换中又分为两种：报文存储转发交换与报文分组存储转发交换。

(1) 线路交换方式

线路交换方式来源于电话网络，在两台计算机之间通过网络进行数据交换之前，首先在网络中建立一个为此次通信专用的实际物理线路连接。其通信过程分为三个阶段。

① 线路建立

如果主机 A 要向主机 B 传输数据，首先要通过网络在 A 与 B 之间建立线路连接。A 首先发送含有需要建立线路连接的源地址与目的地址的“连接请求包”。

图9.9中，如果要在节点 A，B 之间建立线路连接，则节点 A 根据目的主机地址，启动路选算法选择下一个节点 C，并向节点 C 发送“连接请求包”。接到连接请求的下一节点同样根据路选算法，选择再下一个节点，并向其发送“连接请求包”，这样建立起 A—C—D—E—B 的物理线路连接。B 如果接受 A 的呼叫请求，则通过已经建立的物理线路连接 B—E—D—C—A 向 A 发送“连接应答包”，完成此次 A 与 B 间数据交换服务的专用物理线路的连接。

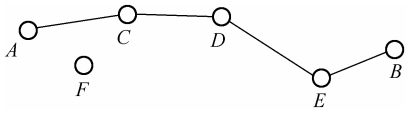


图 9.9 线路的建立与拆除

② 数据传输

在 A 与 B 通过网络的物理线路连接建立以后，A 与 B 就可以通过该连接，实时、双向地交换报文（或报文分组）。

③ 线路释放

数据传输完成后，进入线路释放阶段。一般由 A 向 B 发出“释放请求包”。B 同意结束传输，释放线路后，将向节点 E 发送“释放应答包”。建立的物理连接按 B, E, D, C, A 次序逐站释放，结束此次通信。

线路交换方式中，节点是电子或机电结合的交换设备，完成输入线路与输出线路的物理

连接。交换设备和线路分模拟通信和数字通信两类。节点交换设备不存储数据，不能改变数据内容，不具备差错控制能力。

线路交换方式的优点是：通信实时性强，适用于交互式通信。缺点是对突发性通信不适应，系统效率低。系统不具有存储数据的能力，也不具备差错控制能力。

(2) 存储转发交换方式

存储转发交换方式与线路交换方式的主要区别是：发送的数据与目的地址、源地址、控制信息按一定格式组成一个数据单元，即报文（或报文分组）。通信站点的通信控制处理器要完成数据单元的接收、差错校验、存储、路选和转发功能。

存储转发交换方式有下列明显优点：

- 线路利用率高：由于报文（或报文分组）可以存储，多个报文（或报文分组）可以分时共享一条节点到节点的通信信道；
- 系统效率高：通信控制具有路选功能，可以动态选择报文（或报文分组）通过通信网络的最佳路径，同时可以平滑通信量；
- 系统可靠性高：报文（或报文分组）在通过每个通信控制时均进行差错检查与纠错处理，以减少传输错误；
- 适应性强：通过通信控制处理器，可以对不同通信速率的线路进行速率转换，也可以对不同数据代码格式进行交换。

存储转发交换方式在计算机网络中得到了广泛使用。存储转发交换方式可以分为两类：报文交换与报文分组交换，其结构如图9.10所示。

报文交换在发送数据时，无论发送数据的长度是多少，都把它作为一个逻辑单元，在发送的数据上加上目的地址、源地址与控制信息，按一定的格式打包后组成一个报文。



图 9.10 报文和报文分组结构

报文分组方法在发送站将一个长报文分成多个报文分组，在接收站再将多个报文分组按顺序重新组织成一个新报文。

采用报文分组交换技术的通信子网称为分组交换网。采取的分布式控制的自适应路由选择技术，是目前公用数据交换网中主要的交换技术。这是因为报文分组长度较短，容易检测传输差错，出错重发花费时间较短，有利于提高存储转发节点存储空间利用率和传输效率。

报文分组交换技术在实际应用中又分为两类：数据报（Data Gram，DG）方式与虚电路（Virtual Circuit，VC）方式。

① 数据报方式的特点是。同一报文的不同报文分组可以由不同的传输路径通过通信网络，到达目的节点时可能出现乱序、重复和丢失现象；每一个报文分组在传输过程中都必须带有目的地址和源地址，报文传输延迟较大，适用于突发性通信，不适用于长报文、会话式通信。

② 虚电路交换方式试图将数据报方式与线路交换方式的优点结合起来，达到最佳数据交换效果。其特点如下：

- 在每次报文发送之前，需要在发送方和接收方建立一条逻辑连接的虚电路，一次通信中所有报文分组都从这条逻辑连接的虚电路上通过，因此报文分组不必带目的地址、源地址等辅助信息，报文分组到达目的节点不会出现丢失、重复和乱序的现象。
- 报文分组通过每个虚电路的节点时，节点需要做差错检测，而不需要做路径选择。
- 通信网络中每个节点可以和任何节点建立多条虚电路连接。

(3) 不同交换方式的适用场合

线路交换适合于高负荷的持续通信要求，尤其是会话式通信与语音、图像通信，不适用于突发性通信。报文交换适合于长报文，无实时要求的通信，不适合会话式通信。数据报方式适合于灵活的突发性短报文传输，不适合会话式和有实时性要求的通信。虚电路方式既适合于定时、定对象、长报文通信，也适合会话式通信和语音、动态图像和图形通信要求。

5. 差错控制

(1) 差错控制策略

为了减少通信传输错误，差错控制的主要策略有：

① 纠错码方案。为了能在接收端发现并自纠正传输错误，每个传输的报文分组需要带上足够的冗余信息，但较强的纠错能力，要求附加的冗余码比基本信息码多，因而纠错位数有限，传输效率低，实现复杂，造价高，在一般的通信场合中不易被采用。

② 检错码方案。报文分组仅包含足以使接收端发现差错的冗余信息，但不能确定是哪一比特出错，也不能自动纠正传输差错，需要通过重传机制达到纠错。由于原理简单，实现容易，编码与解码速度快，因而得到了广泛使用。

③ 混合方案。综合上述两种方式，发送端发送的信息码不仅能发现错误，而且还具有一定的纠错能力。接收端收到该码后，如果错误位数在纠错能力以内，则自动地进行纠错；如果错误过多，则要求发送端重发，直到正确为止。

(2) 常用的检错码

① 奇偶校验码是最常见的检错码，方法简单，但检错能力差，一般只用于通信要求较低的环境。分为垂直奇（偶）校验、水平奇（偶）校验和水平垂直奇（偶）校验（即方阵码）。

② 循环冗余编码（Cyclic Redundancy Code, CRC）。检错能力强，实现容易，是目前应用最广的检错码编码方法之一。CRC 码检错方法是将要发送的数据比特序列作为一个多项式 $f(X)$ 的系数，在发送方用收发双方预先约定的最高幂值为 k 的生成多项式 $G(X)$ 去除，求得一个余数多项式。将余数多项式加到数据多项式之后发送到接收端。接收端用同样的生成多项式 $G(X)$ ，去除接收数据多项式 $f'(X)$ ，得到计算余数多项式。如果计算余数多项式与接收余数多项式相同，则表示传输无差错，否则表示传输有差错，由发送方重发数据，直至正确为止。在实际网络应用中，用软件或硬件方法都可以实现 CRC 校验码生成与校验，已有很多超大规模集成电路的通信芯片可以在内部非常方便地实现标准 CRC 校验码的生成与校验功能。

(3) 反馈重发纠错

数据通信系统通常采用反馈重发（Automatic Request for Repeat, ARR）方法来纠正传输错误（如图9.11所示）。实现方法有两种。

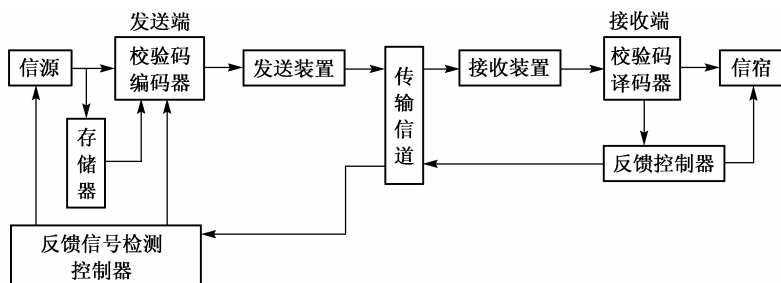


图 9.11 反馈重发纠错实现方法

① 停止等待方式。发送方在发送完一数据帧后，要等待接收方表示已正确接收的应答帧，然后才发送下一数据帧。停止等待方式协议简单，但系统通信效率低。

② 连续工作方式。发送方连续向接收方发送数据帧，同时接收应答帧。一旦得知传输出错，即停止当前数据帧的发送而重新发送，系统通信效率较高。该方式又分为从出错数据帧开始重发的拉回方式与只将出错数据帧重发的选择重发方式两种。

9.1.4 网络协议及其层次结构

网络的发展要求实现不同厂家生产的计算机设备之间的互连操作：既可以用不同厂家的产品来构成系统，也可以在相同功能的不同厂家产品之间实现互换。这样做的前提是开放性——有公开而统一的规范描述及其实现方法，有对标准的共同认识与遵从。

国际标准化组织 ISO 提出的 OSI（Open System Interconnection）参考模型是为实现开放系统互连而建立的，其分层模型如图 9.12 所示，目的是为异种计算机互连，协调进程间通信标准，提供一个共同的基础和标准框架，并为保持相关标准的一致性和兼容性提供共同的参考。

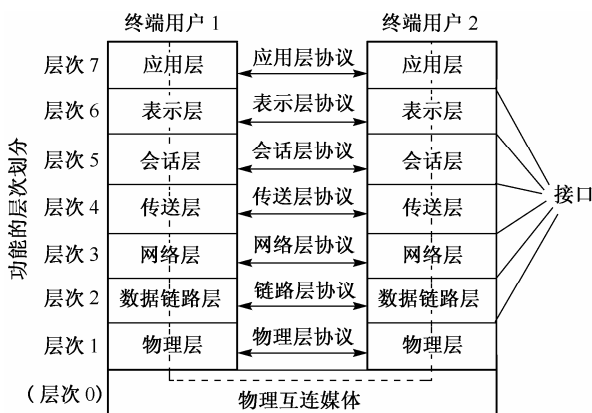


图 9.12 OSI 开放互连参考模型

只有按照 OSI 标准生产的网络产品才能实现开放系统互连。

分层设计方法使得网际联网、网络配置间的连接能容易地实现，这对计算机通信和网络十分重要。利用对应于某分层结构的协议，在系统的逻辑成分间建立有效的联系而不影响其他层。某一层的修改不会破坏整个设计，不会影响其他层（包括相邻层）的工作。例如，网络的传输方式可从并行改为串行，而不影响除最低层外的其他层。

每层完成的功能不同，相邻层之间通过接口发生联系。接口是包含在信息处理系统中同

一节点内部的不同功能层次之间有关的一组规则。同一层中的实体借助于该层协议相互通信，使得整个交互过程规范化。

OSI 参考模型分层的原则是将相似的功能集中在同一层内，功能差别较大时，则分层处理，每层只对相邻的上下层定义接口。每一层的功能是独立的，下一层向上一层提供服务，即提供通信功能和层之间的会话规定，而与其他层的具体实况无关。通常，第 1~3 层功能称为低层功能（LLF），即通信传送功能，这是网络与终端均需具备的功能。第 4~7 层功能称为高层功能（HLF），即通信处理功能，通常需由终端来提供。

1. 物理层（第 1 层）

物理层是开放系统中利用物理媒体实现物理连接的功能描述和执行连接的规程，它提供用于建立、保持和断开物理连接的机械的、电气的、功能的和过程的条件。物理层提供有关同步和比特流在物理媒体上的传输手段，例如，接插件的型号，每根线的作用，1 电平和 0 电平的表示，1 比特的宽度，传输能否在两个方向同时进行，如何进行初始的连接，如何拆除连接等。RS 232C，RS 422A，RS 423A，RS 485 等均为物理层协议。

2. 数据链路层（第 2 层）

数据链路层用于建立、维持和拆除链路连接，将可能有差错的物理链路改造成对于网络层来说是无差错的传输线路。该层对连接相邻的通路进行差错控制、数据成帧、同步等控制。同步数据链路控制（SDLC）、高级数据链路控制（HDLC），以及异步串行数据链协议都属于此范围。

3. 网络层（第 3 层）

网络层规定了网络连接的建立、维持和拆除的协议。它的主要功能是控制网络上信息的切换和路由选择，规定分组（第 3 层的信息单位）在网络中如何传输。

4. 传送层（第 4 层）

传送层确保高质量的网络服务。其重要功能如下：

- (1) 控制开放系统之间数据的收发，确认端到端的数据的完整性，完成开放系统之间的数据传输控制。
- (2) 弥补各种通信网络的质量差异，对经过下三层之后仍然存在的传输差错进行恢复，进一步提高可靠性。
- (3) 通过复用、分段和组合、连接和分离、分流和合流等技术措施，提高吞吐量和服务质量。

5. 会话层（第 5 层）

会话层的主要功能是按照在应用进程之间的约定，正确地顺序收发数据，进行各种形式的对话。

控制方式可以进行如下归纳：

(1) 交替改变发信端的传送控制

为了在会话应用中易于实现接收处理和发送处理的逐次交替变换，设置某一时刻只有一端发送数据。

(2) 单方向传送大量数据时的标记控制

例如，可以将长文件分页发送，当收到前一页的接收确认后，再发下页的内容；当出现意外时，由标记处重发。

6. 表示层（第 6 层）

提供字符代码、数据格式、控制信息格式、加密等的统一表示，把应用层提供的信息变换为能够共同理解的形式，而不改变其内容本身。

7. 应用层（第7层）

实现应用进程（如用户程序、终端操作员等）之间的信息交换。规定在不同应用情况下所允许的报文集合和对每个报文所采取的动作。解决数据传输完整性、发送/接收设备的速度匹配等问题。

9.2 分布式控制系统（DCS）

早期的分布式控制系统（Distributed Control System, DCS）或集散控制系统局限于生产控制领域，如石油化工工业。近年来，分布式控制系统大大扩展了分散控制和集中操作管理的内容。将多种监测控制、计划管理功能有机地集成起来，最终将实现控制自动化和信息管理自动化相结合的计算机集成制造系统（Computer Integrated Manufacturing System, CIMS）。

9.2.1 DCS结构与特点

1. DCS 系统的层次功能

按划分，可以将分布式系统分为 4 级，如图9.13 所示。

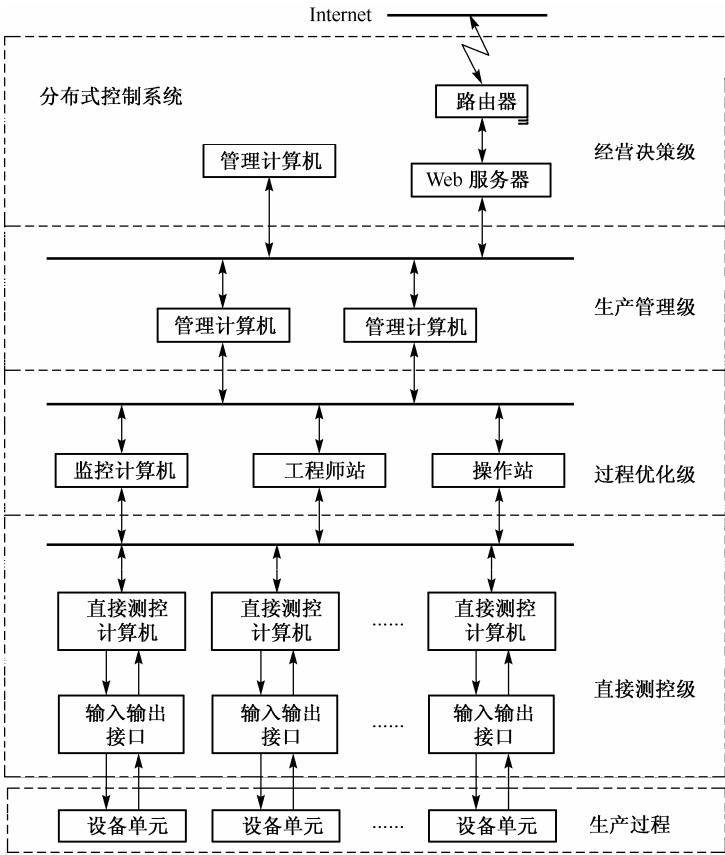


图 9.13 分布式控制系统

(1) 直接测控级

直接测控级是 DCS 的基础，直接面向生产过程，完成各类现场设备的监测、控制，进行

数据采集、数据预处理、开环或闭环控制、设备单元的监测及故障诊断、事故报警等工作，同时还接受和执行上级指令，并向上级传递实时信息。

直接测控级的输入信息是来自各种传感器（如温度、压力、液位变送器）的模拟量信号及开关量信号，其输出是驱动执行机构。

构成直接测控级的主要装置有各种嵌入式测控装置，如工业控制机、单片机系统、可编程程序控制器、智能调节器和其他测控装置。

(2) 过程优化级

综合监视直接控制级中各站点的所有信息，集中显示，集中操作，进行各控制回路的组态、参数设定，以及修改、优化控制过程等。

这一级主要有监控计算机、操作站、工程师站。

(3) 生产管理级

生产管理级为管理者提供针对车间和工厂生产的有关信息，以作出关于生产计划、调度和管理方案，使计划协调和生产管理处于最佳状态。

(4) 经营决策级

经营决策级担负整个企业或集团的总体协调、计划管理、市场营销、预测和决策。与财务、人事、档案、仓库、经营等各办公自动化系统连接，制定长期发展规划和近期的发展计划，综合生产的计划、管理、销售、订货，了解与分析市场经营动向和本部门的财务收支及预算，进行总决策。向下级下达任务，实现总调度。连接到 Internet 上，可进行各种电子商务活动。

经营决策级一般要求数据处理和科学计算能力强、存储容量大的大中型计算机。

DCS 系统中，各级之间和级内各装置之间都由通信网络进行通信联系。系统的有效性和柔性极大程度上取决于通信网络的有效性。

2. DCS 的特点

(1) 硬件和软件模块化

DCS 采用模块化组装结构，适用于不同的设备和不同的控制对象，具有良好的通用性，可以方便地配置构成多级控制系统，并能灵活地扩充、修改和升级。当应用对象改变时（例如，生产工艺要求有改变，或系统应用于另一种生产过程），只需增减某些硬软件模块，就可以扩大或缩小系统的规模，满足对象的要求。因而，有利于企业分批投资，逐步形成一个在功能上和结构上从简单到复杂、从低级到高级的现代化管理系统。

用户只需按照系统的控制方案，以填表的方式来定义选用的软功能模块，就可进行 DCS 的组态，极大地提高了系统设计的效率。软件功能模块包括控制软件包、操作显示软件包、报表打印软件包等，以及自行开发的高级应用软件。

控制软件包为用户提供各种过程控制的功能，主要包括数据采集和处理、控制算法、常用运算式和控制输出等功能模块。这些功能通常固化在现场控制装置中。

操作显示软件包括为用户提供丰富的人机接口联系功能，在取代常规模拟仪表盘的操作站上进行集中操作和监视。可以选择多种 CRT 显示画面，如总貌显示、分组显示、回路显示、趋势显示、流程显示、报警显示和操作指导等画面，并可以在 CRT 画面上进行各种操作。

报表打印软件包可以向用户提供每小时、班、日、月工作报表，打印瞬时值、累计值、平均值、打印事件报警等。

一个优良的计算机监测控制系统应当是灵活协调的。设计时指标要留有一定余地，软硬件设计都应规范化、模块化，便于修改。模块要易于连接和组织。

(2) 通信网络的应用

通信网络是分散型控制系统的神经中枢，它将物理上分散配置的多台计算机有机地连接起来，实现了相互协调、资源共享的集中管理。通过高速数据通信线，将现场控制站、局部操作站、监控计算机、中央操作站、管理计算机连接起来，构成多级控制系统。

DCS 采用统一的系统总线结构和标准化的通信规范。通信距离按用户要求可从十几米到十几千米。通信速率可满足实时控制、企业管理和数据通信的需要。

(3) 运行可靠，使用方便

DCS 在系统结构、冗余技术、自诊断功能、抗干扰措施、器件与部件的性能等各方面，以及设计、安装、调试、运行整个过程的综合考虑，保证了在相对恶劣环境中运行的高可靠性。

DCS 的使用方便包含可操作性、可诊断性和可维护性。操作清晰简明，尽可能降低对操作人员专业技术知识的要求；在故障出现时，能及时定位并排除；采用标准零部件，便于维护更换。

(4) 性能/价格比高

设计、安装、调试周期短，系统的开发费用低，可以根据应用对象的要求和当前的市场情况进行具体的分析与选择。

然而，DCS 也有十分明显的缺点：

- 多级主从关系结构，底层相互间进行信息传递也必须经过主机，从而造成主机负荷过重，效率低下。主机一旦发生故障，整个系统就会“瘫痪”。
- 数字-模拟混合系统，DCS 的现场仪表仍然使用传统的 4~20 mA 电流模拟信号，传输可靠性差，成本高。
- 各厂家的 DCS 自成标准，通信协议封闭，极大地制约了系统的集成与应用。

9.2.2 直接测控级

1. 基本功能

(1) 数据采集功能

对过程参数包括各类热电偶信号、热电阻信号、压力、液位、流量等信号，进行数据采集、变换、处理、显示、存储、趋势曲线显示、事故报警等。

(2) DDC 控制功能

接收现场的测量信号，求出设定值与测量值的偏差，并对偏差进行控制规律的算法运算，在求出新的控制量后，转换成相应的输出信号送至执行机构。

(3) 顺序控制功能

根据来自过程状态输入输出信号和反馈控制功能等状态信号，按预先设定的顺序和条件，对控制过程的各阶段进行逐次控制。

(4) 信号报警功能

以声音、光信号或 CRT 屏幕显示颜色变化等，对超限值、非法状态和事故等进行报警。

(5) 打印报表功能

定时打印报表，随机打印过程参数。自动记录、打印事故报表。

(6) 数据通信功能

完成分散过程控制级与集中操作监控级之间的信息交换。

2. 现场控制站的工作方式

(1) 控制模式。

控制模式的优先级由高到低的顺序依次是硬手动、软手动、自动、串级、上位机。

① 硬手动模式。将现场控制切换到模拟显示操作单元，由操作者操作该单元直接输出控制值。

② 软手动模式。用软件实现手动操作，利用键盘改变阈值，在 CRT 上可显示出输出阈值。

③ 自动模式。处于自动工作状态，给定值由本地给定，由偏差控制算法运算后输出控制量。

④ 串级模式。将主回路的输出值，作为送到副回路的给定值。

⑤ 上位机模式。由上位机设定给定值或根据高级控制算法的运算结果自动修改给定值，然后传送给现场控制装置，完成优化控制。

(2) 无扰动切换操作

在控制系统中，工作模式的切换，可能会引起输出阈值的突然变化，产生扰动。

从自动到手动方式的无扰动切换要求，模拟显示操作单元显示自动模式的输出阈值和手动输出阈值，切换之前，将手动输出阈值与自动阈值调到相同，然后再换到手动模式。

从手动到自动方式的无扰动切换则要求采取以下措施：

① 测量值（PV）跟踪。在各种手动模式下，把测量值送给给定值，保证偏差为零，输出值则不受影响。

② 阈值跟踪。采用阈值反馈的办法，将手动输出阈值自动反馈，输入到现场控制站的输入通道，作为一个检测点信号输入，经过转换及变换后送到模拟量输出通道，此时的输出值正好跟踪手动输出的阈值。

③ 工作模式判定。根据自动、手动、串级、上位机等工作模式的相应标志状态进行模式判定。

3. 分散过程控制级的系统构成

分散过程控制级含有多种类型的现场控制装置和通信控制单元，在构成系统时还应注意以下问题。

(1) 机箱（柜）

现场控制装置的机箱（柜）内部均装有多层机架，以供安装电源及各种部件之用。其外壳均采用金属材料，活动部分之间应有良好的电气连接，为内部的电子设备提供电磁屏蔽。为保证电磁屏蔽效果和操作人员的安全，机柜要求可靠接地，接地电阻应小于 $4\ \Omega$ 。

机箱一般均装有风扇，以提供强制风冷气流保证电子设备的散热降温，为防止灰尘侵入，在与箱（柜）内进行空气交换时，最好采用正压送风将箱（柜）外低温空气经过过滤网过滤后压入箱（柜）内。

(2) 电源

电源工作的可靠性对系统至关重要。经验表明，电源故障往往占系统电子器件故障的一半。作为现场控制装置稳定的重要保证，电源一方面要能承受通过静电放电和通过传导方式进入的外界浪涌和快速脉冲群的干扰，另一方面也必须限制其本身对其他设备的干扰。

根据国标 GB9813—88“微型电子数字计算机通用技术条件”规定，供电电压的变化应为 $(220 \pm 22)\text{ V}$ ，电源频率的变化应为 $(50 \pm 1)\text{ Hz}$ 。国标 GB/T17618—1998“信息技术设备抗干扰度限值和测量方法”规定了计算机设备电源输入端口的抗扰度，电源上的噪声不允许

超过设备的抗扰度，否则就可能使计算机性能降级，甚至损坏设备。标准规定，对于电源中的快速瞬间脉冲群，上升时间为 5 ns，重复频率 5 kHz，持续 15 ms，其尖峰脉冲幅值不能超过 1 kV。对于电源中的浪涌，上升时间为 2 μs，脉宽为 50 μs，其尖峰幅度不能超过 1 kV（电源线之间浪涌）或 2 kV（电源线与地线之间浪涌）。

目前广泛采用的直流稳压开关电源，对输入电压的适应范围大，也具有相当好的抗干扰能力，但由于其固有的工作原理和特性也可能产生高频干扰。因此，在设计、采购与使用时都应当注意到电源必须有良好的滤波与噪声抑制措施。

应当有可靠的稳定供电电源和防止掉电的掉电检测与保护措施，例如，采用不间断电源（UPS）。

现场控制站内各功能模板可能需要多种直流电源，如+5 V、±15 V（或±12 V）、+24 V 等。一般说来，在电气上将对主机供电的电源与现场测控装置或执行机构供电的电源互相隔离，可以减小相互干扰。

(3) 接地

现场控制装置中，接地的一个重要目的是为系统各部分提供稳定的基准电位。从电磁兼容性试验结果看，系统的可靠性与系统的接地正确与否关系极大。

现场控制装置中的“地”通常有多种含义：

① 机壳地，即“大地”，因为通常屏蔽线的屏蔽层都接机柜外壳与大地相连，又称为“屏蔽地”。每个插件箱上的屏蔽地与电源 0 V 基准地必须分开，屏蔽地接到插件箱机壳或接地母排上。电源插件外壳需通过其引脚连到屏蔽地，网卡的面板需连到屏蔽地。外部信号的屏蔽层原则上接到机柜中的接地母排上。系统总线屏蔽层分段一点接地，各插件箱外壳及接地母排都应有专用导线连到机柜骨架上，屏蔽层只能一端接地以免形成环流。

② 主机电源地应与机壳地分开。在需要对主机电源开关信号进行采样的设备中，数字接口电路应通过光电耦合器与电源开关信号相连。

③ 直流电源基准地，即直流低压电路（+5 V，±15 V，±24 V）的 0 V 基准导体。每台设备的电源 0 V 基准导体必须具有足够的截面，或是在母板上将各插件的 0 V 线连通，或是通过插件引脚连到 0 V 的汇流条上。可根据屏柜的电磁兼容性试验结果，决定各设备的电源 0 V 基准地是互相独立的，还是通过导线互连的。

④ 通信信号地。对于隔离后的 RS-485 或 RS-422 的通信电路低压地，应采用第三导体互连。第三导体应与双绞屏蔽线一起制成一根电缆。

(4) 手动/自动显示操作单元

作为后备安全措施，手动/自动显示操作单元可以显示测量值、给定值、自动和手动输出阈值，并具有硬手动操作功能，可直接调整输出阈值。

9.2.3 过程优化级

过程优化级的主要功能是通过显示操作站完成显示操作和进行控制系统的生成、组态。显示操作站收集各个现场测控装置的数据，将过程参量的信息集中化，完成显示、操作、记录、报警等功能。通过简单的操作，显示过程量、各种工艺流程图、趋势曲线，改变过程参数，如设定值、控制参数、报警状态等。

1. 显示操作站的构成

显示操作站功能强、速度快、记录数据量大，大多采用工业 PC 机作为监控计算机，其键

盘有两种类型：操作员键盘和工程师键盘。

(1) 操作员键盘

操作员键盘面向操作工，力求操作方便、直观、易掌握。多采用有防水能力、防尘能力、有明确图案标志的薄膜键盘。可在键体内装有电子蜂鸣器，以提示报警信息和操作响应，也可采用触摸屏显示输入。

(2) 工程师键盘

工程师键盘供系统控制工程师编程和组态用，通常采用标准键盘。

2. 显示操作站的功能

显示操作站为用户提供仪表化的操作环境，通过 CRT 操作来实现整个分布式控制系统的高效率运转。它是显示操作的中心，也是信息集中分配中心。具备运转整个系统需要的集中监视操作、报警、显示、报表、通信等操作员功能和控制系统生成、维护需要的工程师功能。

(1) 操作员功能

正常运行时的工艺监视和运行操作，主要由各种指示画面构成。

① DDC 标准三画面

- 总貌画面：用来监视分布式控制系统的运行状态，可指示出仪表的位号，测量的光柱图形。当一些运行状态发生异常时，相应的仪表位号指示变色、闪光、声音报警。
- 组貌画面：每幅显示有若干个 DDC 回路。操作对应键，就可调出相应的组貌画面。
- 回路画面：显示单个 DDC 回路的各参数，也称为调整画面，可进行控制参数的修改、设定，完成单个回路的控制作用。处于串级状态下的回路，可在画面上同时显示出相关回路的状态及参数。

② 流程显示画面

工艺流程图由二部分组成：

- 静态图：不随工况变化的基本图形；
- 动态图：随工况而变化的数据、曲线、图形等，并且定时刷新其变化部分。

③ 趋势曲线画面

对过程参数的变化趋势进行记录、显示、存储，在一幅画面以不同的颜色显示多条曲线；当曲线较多时，可以分组登记、分组显示。

趋势曲线可被存储起来，进行逐天、逐月的历史趋势显示，有助于分析和管理；通过趋势曲线的变化，可以及时掌握过程变化情况，也可以分析控制系统运行状态，分析事故和质量波动的原因。

④ 操作指导画面

操作员按画面指导，可以准确及时地进行操作，保证工序正常进展，还可以拒绝错误操作，保证安全生产。

⑤ 报警画面

按时间顺序记录过程数据发生、上下限报警和设备异常等，并且用颜色变化、闪光、文字和声响来区分报警级别，以引起操作员注意，立即采取相应措施。

(2) 工程师功能

工程师功能主要包括系统的组态、控制、维护和管理等。

- ① 系统组态：系统工程师可依照给定的运算功能模块进行选择、连接、组态和设定参数，构成一个应用系统，而无须用户编程序。
- ② 系统控制：依据组态执行相应的控制功能。
- ③ 系统维护：定期检验各个作业，且在系统的建立、修改时保存内容。
- ④ 系统管理：管理系统的监视功能、控制功能，以及制表、事故报警、打印等。

9.2.4 生产管理、经营决策和CIMS

从企业整体来看，DCS 是综合生产系统的基础。生产管理级和经营决策级都是管理信息系统（Management Information System, MIS）的有机组成部分。

企业生产的各个环节，即从市场分析、产品设计、加工制造、经营管理到售后服务的全部生产和经营活动，是一个不可分割的整体，必须紧密连接，统一考虑。

企业的全部生产和经营过程实质上是一个数据采集、传送和加工处理的过程，最终形成的产品可看做是数据的物质表现。

以计算机网络为支撑的 CIMS，可以实现同级各自动化子系统的协调配合和上下级间自动化子系统的递阶控制，这个系统将传统制造技术与信息技术和管理技术相结合，利用计算机软硬件、网络和数据库技术，将企业的经营、管理、计划、产品设计、加工制造、销售及服务等部分与人、财、物集成起来，保证高效率、高质量和高柔性地管理企业，提高竞争能力（如图 9.14 所示）。

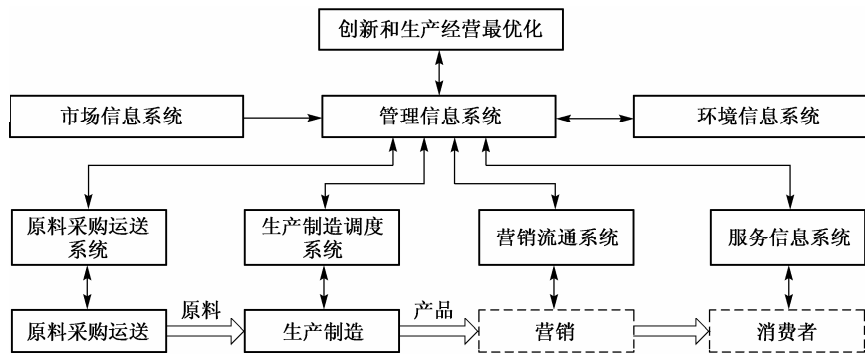


图 9.14 管理信息系统

1. CIMS 的内容

(1) 计算机辅助设计（Computer Aided Design, CAD）

利用计算机直接对产品的参数、结构进行分析，根据国家标准规范和计算数据，自动地绘制图纸，以及修改、审查产品设计。可以缩短产品设计周期，进行产品的多方案选择，实现复杂产品、单件小批量产品的设计。

(2) 计算机辅助制造（Computer Aided Manufacturing, CAM）

利用计算机把设计的信息转化为产品，进行生产设备的管理、控制和操作。可以控制机器的运行，处理产品制造中所需要的数据，控制材料的流动，测试检验产品性能。能提高产品质量，降低成本，改善人员工作条件。

(3) 计算机辅助工艺计划（Computer Aided Process Planning, CAPP）

利用计算机设计产品的工艺流程，动态地更新设计的工艺、工序、工时，保证产品质量，

节省工时和降低成本。

(4) 计算机辅助管理

利用计算机对经营管理过程中的信息进行加工处理,以低费用、快速度、高精度和强可靠性提供管理信息,辅助各级管理人员进行管理和决策,提高管理水平和经济效益。

计算机辅助设计、辅助制造、辅助工艺计划这三者都是以工程为对象的实体系统,计算机辅助管理是以事务管理为对象的软体系统,以往它们所属的制造部门和管理部门两者之间没有直接的技术上的联系和信息交流,而在 CIMS 系统中,计算机充分发挥其作用,在属于不同的应用领域的各个子系统之间实现信息高度共享,把企业融为一个有机整体。

2. CIMS 的关键

CIMS 是一个高集成度的系统,其关键是管理的集成和计算机应用技术的集成。

任何一个企业都是一个整体,是一个围绕着如何达到企业生产经营目标的系统。系统的整体效益大于各部分效益之和,这是因为整体所产生的功能是一种质的变化,是各部分不具备的功能,不能用各部分功能的简单相加所代替。无论是设计、制造、工艺或者管理,都是企业系统的一个组成部门,都必须服从系统整体需要,相互联系、相互影响,环环紧扣,组成一个有机的整体,建立企业的生产管理过程,共同完成生产经营活动的目标。

企业的生产经营活动是围绕着产品而展开的。企业通过销售部门接受来自用户方面的需求,结合本公司技术和人员的条件,运用计算机辅助设计(CAD)技术,进行产品的设计工作。提供产品的全套图纸和技术参数,给出产品结构和材料的要求,建立零件数据库、算法库、产品结构数据库,通过计算机网络系统将产品信息传递给计算机辅助工艺计划(CAPP)系统。CAPP 系统根据传送来的产品信息完成工艺分析、工艺编制、工艺装备设计和制造、工艺准备,并建立产品工艺数据库、算法库等。通过计算机网络系统还可以将零部件信息传送给计算机辅助制造(CAM)系统,产品正式进入生产过程。根据 CAD 和 CAPP 传送的信息,建立物料需求计划、采购计划。根据材料、工时、设备的情况平衡生产能力,然后下达生产计划和进行车间生产的调度、零部件入库、装配、发运等工作。这个过程涉及生产中的物料管理和信息管理。

物料管理包括由原材料至最终产品的管理和加工设备的管理。

信息管理包括:对整个产品生产全过程要进行计划安排、能力平衡和生产调度,这是生产计划和调度方面的信息管理;对在制品和制成品的检验涉及质量信息管理;生产过程同时也是一个物料消耗过程,因此要对生产中所用的物料采购、存储信息进行管理,这又涉及物流信息管理;为了保证生产的顺利进行,还需要对所用设备及其运行状态进行设备信息管理;对生产过程所消耗的能源也需要进行管理,所以又产生了能源信息管理;在以上各过程中又都有资金费用的发生,因此又涉及会计成本信息和资金信息管理;而各种信息经汇总成为统计分析所需要的内容,产生了统计信息管理。以上所有这些管理信息组成了管理信息系统(MIS)。MIS 的输出信息是生产过程的动态信息,如生产中实际工时、材料实际采购数量、存储状况、计划执行情况等,这些信息是生产过程的动态反映,计算机辅助管理所建立的 MIS 将这些信息及时地通过计算并反馈给 CAD, CAM 和 CAPP,促使这些系统的进一步优化,使整个企业由静态管理走向动态管理。

如图9.15所示,在 CIMS 中,在共享数据库子系统和网络子系统的支持下,管理信息子系统、技术信息子系统、制造自动化子系统和质量信息子系统之间,物质运动紧密结合,信息传递关系密切,组成了从工厂订单——设计——工艺——计划——制造——装配——运输等一

系列物质运动和信息交往的系统。

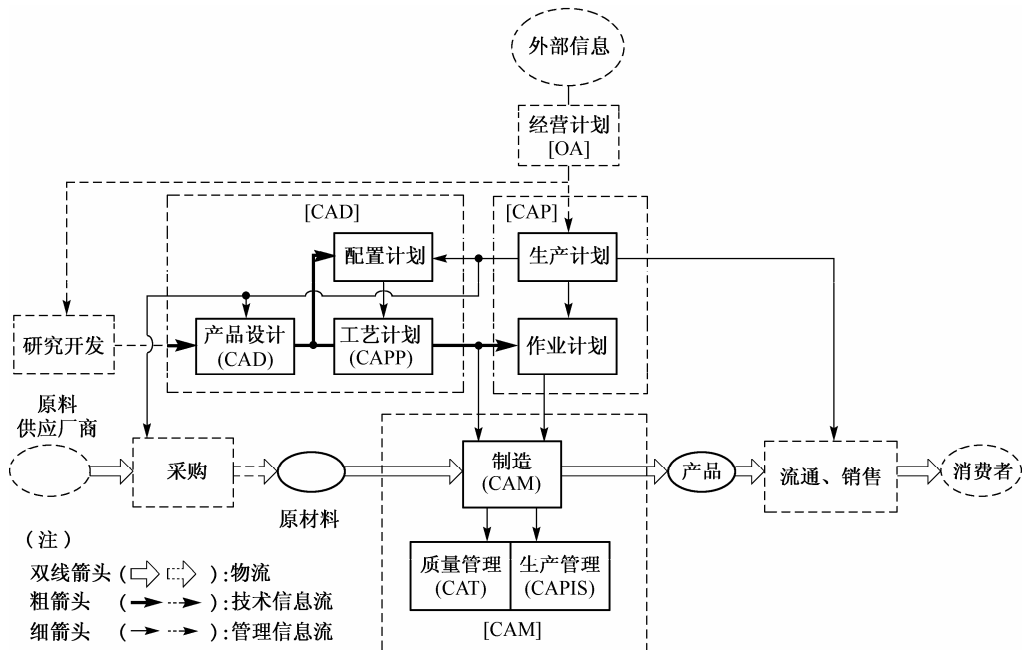


图 9.15 CIMS 结构框图

9.3 现场总线控制系统（FCS）

现场是指工作环境处于控制系统底层的设备，现场总线（Fieldbus）是连接现场设备和自动化系统的数字式、双向传输、多分支结构的通信网络。

现场总线控制系统（Fieldbus Control System, FCS）打破了传统控制系统中，位于现场的测控设备与位于控制室的控制器之间，分别一对一物理连接的结构形式。

作为控制设备级的基础通信网络，现场总线与一般的计算机数据通信网不同之处是，网络负载稳定，多为短帧传送，信息交换频繁；要求可靠性高，协议简单，容错能力强，实时性强，有一定的时间确定性，安全性好，成本低，可以沟通与更高控制管理层次之间的联系。

FCS 作为新一代控制系统，一方面，把 DCS 的集中与分散相结合的集散系统结构，变成了把控制功能彻底下放到现场的新型全分布式结构；另一方面，突破了 DCS 系统采用通信专用网络的局限。

9.3.1 现场总线的技术特点与优点

1. 现场总线的技术特点

(1) 分散性

现场总线系统将传感测量、补偿计算、工程量处理与控制等功能分散到现场设备中完成，仅靠现场设备即可完成自动控制的基本功能，并可随时诊断设备的运行状态，构成一种新的全分布式控制系统，从而简化了系统结构，提高了可靠性。

(2) 数字通信

现场总线系统可实行点对点，一点对多点的数字通信，是专为在现场环境工作而设计的。

它可支持双绞线、同轴电缆、光缆、射频、红外线、电力线等通信介质（如图9.16 所示），具有较强的抗干扰能力。可以实现在一对电线上传输多个信号（包括多个运行参数值、多个设备状态、故障信息），也可以为多个设备提供电源，并可满足本质安全防爆要求。

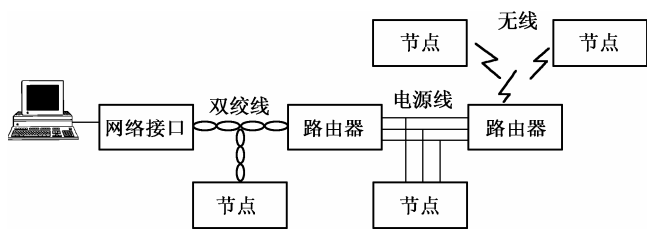


图 9.16 可支持多种通信介质的现场总线控制系统（FCS）

(3) 开放性

现场总线系统的开放性是指通信协议公开和相关标准的一致性，只要遵守相同总线标准，不同厂家的设备或系统之间都可以进行互连并实现信息交换。用户可以根据自己的需要和考虑，把来自不同厂家的产品组成大小随意的系统，还可对性能类似的设备进行互换而实现互用。也就是说，现场总线系统的开放性把控制系统集成的主动权交给了用户。

2. 现场总线的优点

(1) 提高了系统的准确性和可靠性

由于现场总线设备都是以微机为核心的设备，与模拟信号相比，从根本上提高了测量、控制的准确性，而设备功能的强化和系统结构的简化，又大大提高了系统的工作可靠性。

(2) 用户掌握了系统集成的主动权

只要符合总线协议，用户可以自由选择不同品牌的产品，而不必担心兼容性。设计、重构也比较简单。

(3) 降低了系统的投资和复杂性

硬件数量与投资、安装费用和维护开销都可大幅度节约。由于一对双绞线或同轴电缆就可挂接多个智能设备，功能单元和连线大大减少，维护也大大简化。

9.3.2 现场总线的调度策略

为了满足现场总线控制系统对实时性的要求，采用了多种调度策略。例如：

FF 现场总线中采用了令牌总线网，即在物理上是总线网而在逻辑上是令牌网，综合了总线网接入方便、可靠性好和令牌网“无冲突”、时延性好的优点，还采用了 H1, H2 两条总线的优先级措施。

LonWorks 的 LonTalk 协议使用了一种新的称为 P-CSMA（Predictive-CSMA）的算法。这种算法保留了 CSMA（载波监听多路访问）的优点，而克服了它用于控制时的缺点。所有的 LonWorks 节点对介质的访问都是随机的。当有两个或多个节点同时等待网络空闲，以便发送数据包时，这种算法避免了在其他算法中无法避免的冲突。

在 LonTalk 协议中，节点随机地分布在最少为 16 个随机槽的不同延迟水平上。在 P-CSMA 中，当节点要发送一个报文时，它以固定的概率 P 给出随机时间槽的数量。然而，LonTalk 协议中概率 P 是根据网络的负载来动态调整的。当网络空闲时，所有节点只随机分布在 16 个槽上。当估计到网络上的负载增加时，节点将分布在更多的时间槽上，增加的数量由 n

(范围从1到63)来决定, n 称为信道上积压工作的估计, 代表下一次要发送数据包的节点数。这种对积压工作的估计和动态调整的介质访问方法, 使得 LonTalk 协议在网络负载较轻时提供较少的时间槽数, 在网络负载较重时提供较多的时间槽, 从而在负载较轻时, 使介质访问延迟最小化, 而在负载较重时使冲突的可能最小化。

LonTalk 协议可以有选择地提供优先级机制, 以提高对重要数据包的响应速度。协议允许用户在信道上分配优先级时间槽 (Priority Time Slots), 它专用于具有优先级的节点。优先级槽的网络管理可以保证有且只有一个节点, 在信道上被赋给了一个特定的优先级槽。

这样, 设计的优先级与冲突检测的组合, 使得响应时间是有限的, 有效地解决了时延不确定性问题。

Profibus 中, 主站具有对总线的控制权, 可主动发送信息。对多主站系统来说, 令牌在主站之间按地址编号顺序, 沿上行方向进行传递, 得到令牌的站点可在一个事先规定的时间内拥有总线控制权, 向从站发送或索取信息, 并事先规定好令牌在各主站循环一周的最长时间。

CAN 采用总线仲裁技术, 当出现几个节点同时在网络上传输信息时, 优先级高的节点可继续传输数据, 而优先级低的节点则主动停止发送, 从而避免了总线冲突。

9.3.3 现场总线的OSI模型

现场总线就是自动化领域的开放互连系统。

自 20 世纪 80 年代末以来, 逐渐形成的几种有影响的现场总线技术, 大都以 ISO/OSI 模型作为基本框架, 协议是分层的, 但可以简化, 并根据行业的应用需要在增加某些特殊规定后形成标准, 在较大范围内取得了用户与制造商的认可。几种现场总线开放互连参考模型如图9.17所示。

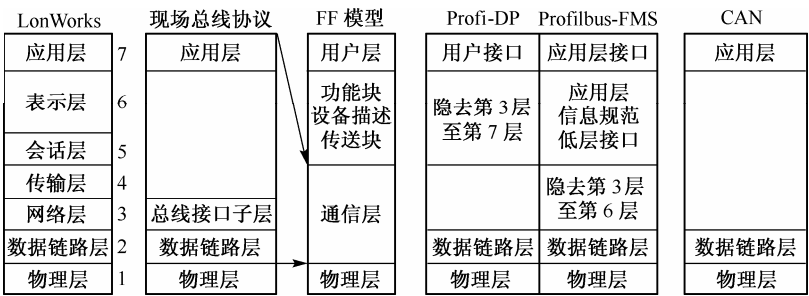


图 9.17 几种现场总线开放互连参考模型

9.4 典型的现场总线

9.4.1 基金会现场总线 (FF)

1994 年 9 月, 以美国 FiSher-Rousemount 公司为首, 联合 Foxboro、横河、ABB、西门子等 80 家公司制订的 ISP 协议, 和以 Honeywell 公司为首, 联合欧洲等地的 150 家公司制订的 World FIP 协议, 两大集团合并成立了现场总线基金会, 致力于开发出国际上统一的现场总线协议——基金会现场总线 (Foundation Fieldbus, FF)。它以 ISO/OSI 模型为基础, 取其物理层、数据链路层、应用层为相应层次, 并在应用层上增加了用户层。在过程自动化领域得到了广泛支持。其通信系统的主要组成部分如图9.18所示。

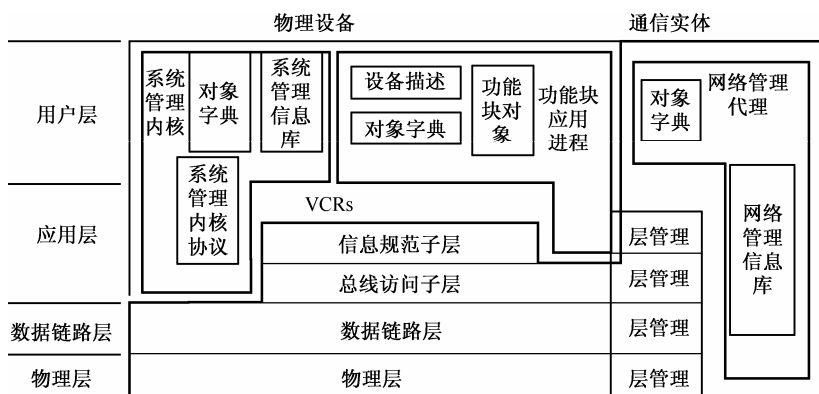


图 9.18 FF 总线通信系统的主要组成部分

1. 物理层 (Physical Layer)

FF 总线的物理层遵循 IEC 1158-2 标准，还涉及系统安装的若干规定：不但规定了物理层的内部参数，还对影响物理信号波形、幅度的相关因素，如媒体种类、传输距离、接地、屏蔽等制定了相应标准。

(1) 传输介质可以采用有线电缆、光纤和无线通信。采用有线电缆作为传输介质的标准为 ISA-SP50.02Port2。

(2) 通过有线电缆传送信号的波特率定义了两种速率标准。

H1: 31.25 kbps 低速率网络。采用 H1 标准，可以利用现有的有线电缆。在同一电缆上连接 2~6 台现场装置，能满足本质安全要求和利用同一电缆向现场装置供电。另行供电，可以连接 2~32 台装置，但不满足本质安全要求。

H2: 1 Mbps/2.5 Mbps 高速网络。大大提高了传输波特率，但不能用信号线供电。

(3) H1 标准最大传输距离为 1900 m (无须中继器)，最多串接四台中继器。H2 标准在 1 Mbps 波特率下，最大传送距离为 750 m；在 2.5 Mbps 波特率下，最大传送距离为 500 m。

(4) 现场总线协议支持总线形、树形和点对点形三种拓扑结构。其中，树形结构仅支持低速 H1 版本。

(5) 编码方式和报文结构

控制设备的数据交换采用同步串行半双工方式，数据信号采用上升沿代表逻辑 0，下降沿代表逻辑 1 的 Manchester 编码方式。数据传输过程是同步的，每帧报文的格式由前同步符 (Preamble)、首定界符 (Start delimiter)、数据段和尾定界符 (End delimiter) 四部分组成。每段报文中数据段的开头和结尾部分由不同于 Manchester 码的非数据正 $N+$ 和非数据负 $N-$ 有效地标明数据段的范围。前同步符和定界符由发送设备中的物理层自动加到报文中，并在接收设备的物理层中被自动去掉。

2. 数据链路层 (Data Link Layer)

数据链路层的任务是保证数据的完整性和决定何时与谁进行对话，由上下两部分组成：下层部分功能是对传输介质传送的信号进行发送、接收控制，上层部分功能是对数据链路进行控制，保证数据传送到指定的装置。

FF 总线采用了将令牌和查询通信方式融为一体的收发控制。在一个网络中可以有几个主站，初始化时，仅允许一个站处于“讲”工作状态。该主站讲完后，通过查询，用特殊的帧结构把“讲”工作状态送给另一主站。

网络中的装置均有不冲突的站地址，所有的帧中包含目标地址和源地址。

为了保证数据传送的可靠性，在发送站的每一帧中加入两个字节的帧检查序列码；在接收站进行解码，判断是否出错。

FF 总线系统有两类信息：工作信息是在装置之间传送的数据，例如，过程变量；背景信息是装置与操作台之间传送的数据，例如，组态和诊断。

3. 应用层（Application Layer）

应用层包括现场总线访问子层（FAS）和现场总线报文规范（FMS）两部分。

现场总线访问子层（Fieldbus Access Sublayer, FAS）提供被称为虚拟通信关系（Virtual Communication Relationships, VCR）的 3 类服务：

- 发布/索取（Publisher/Subscriber）；
 - 客户-服务器（Client/Server）；
 - 报告分发（Report Distribution）。
- 现场总线报文规范（Fieldbus Messaging Specification, FMS）规定了访问应用进程（Application Process, AP）的报文格式及服务；FMS 与对象字典（Object Dictionary, OD）配合，为现场设备规定了功能接口；FMS 通过调用 VCR，在现场设备之间传递报文。

4. 用户层

用户层规定了标准的“块”，供用户组态成为系统，不同的块表达了不同类型的应用功能。FF 总线定义了资源块、10 个基本功能块、19 个先进功能块，以及一组变换块。功能块应用进程中的功能块及其与对象的连接如图9.19所示。

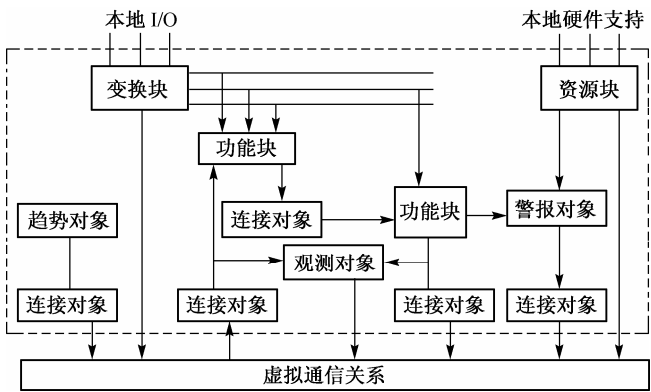


图 9.19 功能块及其与对象的连接

- 资源块：表达应用进程的网络硬件和软件对象，描述了现场总线设备的特征，如设备名、制造者、系列号。一个设备中只有唯一的一个资源块。
- 变换块：按所要求的采样率从传感器中取得数据，并写入相应的硬件之中。
- 功能块：是参数、算法和事件的完整组成，按周期性调度或外部事件驱动功能块的执行，通过算法把输入参数转换为输出参数，实现应用系统的控制功能。输入和输出功能块连接到变换块，与设备的 I/O 硬件相互联系。FF 总线规定的标准功能块如表 9.1 所示。

表 9.1 FF 总线的基本标准功能块

功能块名称	功能块符号	功能块名称	功能块符号
模拟量输入	AI	离散输出	DO
模拟量输出	AO	手动装载	ML
偏置	B	比例微分	PD
控制选择	CS	比例积分微分	PID
离散输入	DI	比率系数	RA

在用户应用中采用了以下对象：

- 连接对象：规定了功能块之间的连接关系，它包括一个设备内部各块之间和不同设备的输入与输出之间的连接关系。
- 趋势对象：收集短期历史数据并存储在一个设备中，提供了回顾其特征的历史信息。
- 报警对象：对状态偏离进行监测。当报警和事件发生时，发出事件通知；如果在预定的时间之内没有收到响应，将重发事件通知。
- 观测对象：支持功能块的管理和控制，将操作数据转换成组并进行处理，使其可被一个通信请求成组地访问，提供了对状态与操作的可视性。

9.4.2 LonWorks

LonWorks 现场总线技术是由美国 Echelon 公司推出并由它与 MOTOROLA、东芝公司共同倡导，于 1990 年正式公布而形成的。它采用了 ISO/OSI 模型的全部七层通信协议和面向对象的设计方法，通过网络变量把网络通信设计简化为参数设置，被誉为通用检测控制网络。该现场总线的通信速率从 300 bps 至 1.25 Mbps 不等，直接通信距离可达 2700 m（78 Kbps，双绞线），节点数最多可达 32385 个（255 个子网×127 节点/子网）。支持双绞线、同轴电缆、光纤、射频、红外线、电源线等多种通信介质，并开发了相应的本质安全防爆产品，即使在不正常条件下，如危险气体杂质混合的空气中，也不会因设备或电缆释放的电能为热能，而引起火花或爆炸。

1. 神经元芯片

神经元（Neuron）芯片（如图9.20 所示）是 LonWorks 技术的核心，LonTalk 协议被封装在神经元芯片中并得以实现；它既是 LON 总线的通信处理器，又可作为采集和控制的通用处理器。LonWorks 技术中所有关于网络的操作，实际上都是通过它来完成的。

(1) 处理单元

集成芯片中有 3 个 8 位 CPU。

MAC CPU：实现介质访问控制（Media Access Control, MAC）与处理，用于完成开放互连模型中第 1 层和第 2 层的功能；

网络 CPU（Network Processor）：用于完成第 3 层至第 6 层的功能，进行网络变量处理的寻址、处理、背景诊断、函数路径选择、软件计时、网络管理、网络通信控制、收发数据包等。

应用 CPU（Application Processor）：执行操作系统服务与用户代码。

(2) 存储器

芯片中存储器实现 CPU 之间的信息传递，并作为网络缓冲区和应用缓冲区。芯片内装有 0.5~2KB RAM 和 1~2KB EEPROM。MC143150 系列支持外部存储器，MC143120 系列则内装 10 KB 容量的 ROM，不支持外部存储器。

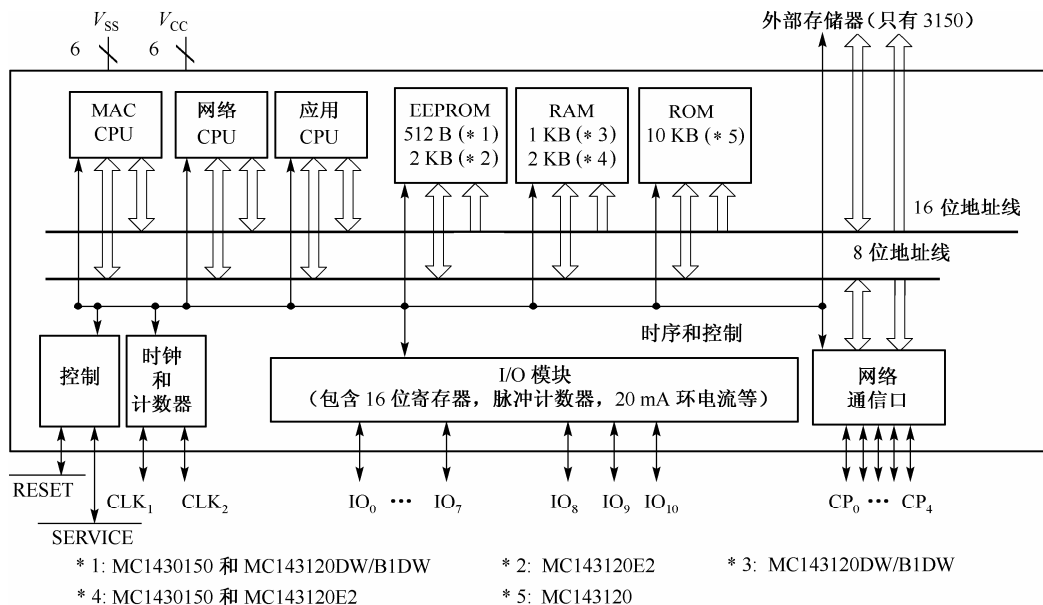


图 9.20 神经元芯片结构框图

(3) 输入/输出

神经元芯片上特设置 11 个 I/O 口，有 34 种预编程设置，可根据不同的需求进行灵活配置，便于同外围设备进行接口。有一个多路选择时间计数器和一个专用时间计数器。

(4) 通信端口

通过不同的收发器（见表 9.2）可以支持双绞线、电力线、无线、红外线、光纤、同轴电缆等多种通信介质。

可以将神经元芯片五个通信管脚配置为单端、差分和专用三种接口模式，以适合不同的编码方案和不同的波特率。

2. 节点与网络

(1) 节点

神经元芯片加上收发器就可以构成一个典型的现场控制节点。

对于复杂的、多回路的控制，可以采用 Host Base 结构：将神经元芯片作为通信协处理器，用高级主机的资源来完成复杂的测控功能，如图 9.21 所示。

表 9.2 神经元芯片的典型收发器类型

收发器类型	波特率
EIA-485	300 bps~1.25 Mbps
自由拓扑型和总结型双绞线带变压器 (可选通过双绞线供 48 V 电源)	78 kbps/1.25 Mbps
电力线（载波）	4 kbps
电力线（扩频）	10 kbps
无线（300 MHz）	1200 bps
无线（450 MHz）	4800 bps
无线（900 MHz）	39 kbps
红外线	78 kbps
光纤	1.25 Mbps
同轴电缆	1.25 Mbps

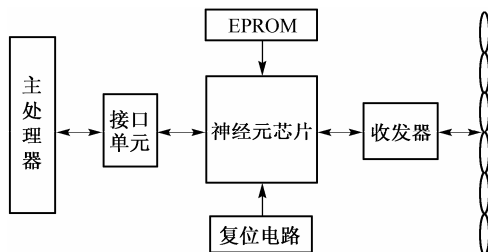


图 9.21 Host Base 节点的结构框图

(2) 路由器

在 LonWorks 技术中, 路由器是中继器、桥接器、配置型路由器和学习型路由器的统称。LonWorks 路由器能支持从简单到复杂的网络的连接, 使 LON 总线突破了通信介质、通信距离、通信速率的限制, 各个节点可以透明地通信, 就如同把它们安装在一个通道上一样。

路由器用于:

- 扩展通道的容量。如使用桥接器来增加多通道以支持更多的节点, 也可以使用中继器延长通道的长度。
- 连接不同的通信介质或波特率。
- 提高 LON 总线可靠性。连到一个路由器上的两个通道在物理上是隔离的, 相连的部分之间相互隔离可以防止因一部分失效而影响其他部分工作。
- 全面提高网络性能。系统内可以用路由器隔离通信。由于大多数节点通信是在某一部分内部进行的, 而不是在各部分之间进行。在各部分之间使用智能路由器可以提高整个网络的吞吐率, 同时也可以缩短通信的响应时间。

(3) 网络管理

网络管理的主要功能有以下三个方面:

- 网络安装。LON 总线通过动态分配网络地址, 并通过网络变量和显示报文来进行设备间的通信。网络安装可通过 Service pin 按钮或手动的方式设定设备的地址, 然后将网络变量互连起来, 并可以设置报文四种方式: 发送无响应、重复发送、应答和请求响应。
- 网络维修。网络维护主要是在系统正常运行的状况下, 增加删除设备, 以及改变网络变量显示报文的内部连接。网络修理包括检测和替换: 检测出设备出错是由于应用层的问题 (例如, 一个执行器由于马达出错而不能开闭), 还是通信层的问题 (例如, 设备脱离网络); 替换出错设备只需从数据库中提取旧设备的网络信息下载到新设备即可, 而不必修改网络上的其他设备。
- 网络监控。提供给用户一个系统级的检测和控制服务, 用户可以在网上, 甚至以远程的方式 (如 Internet) 监控整个系统。通过节点、路由器和网络管理这三部分的有机结合, 就可以构成一个带有多介质、完整的网络系统。

3. LonTalk 协议

LonTalk 是 OSI 参考模型的七层协议的一个子集。它包容了 LON 总线的所有网络通信的功能, 网络操作系统通过所提供的网络开发工具生成固件, 可使通信数据在各种介质中可靠地传输。LON 总线能够直接面向对象通信, 只需通过网络变量的互相连接便可实现节点之间的通信。LonTalk 和 OSI 七层协议的比较如表 9.3 所示。

(1) LonTalk 协议的网络管理和网络诊断

LonTalk 协议提供四类服务:

- 地址分配: 分配所有节点的地址单元, 包括域号、子网号、节点号, 以及所属的组名和组员号, 注意 NeuronID 是不能分配的。
- 节点查询: 查询节点的工作状态及网络通信中的错误统计, 包括通信 CRC 校验错、通信超时等。
- 节点测试: 发送一些测试命令来对节点进行测试。

- 路由器配置：设置路由器的配置表。

表 9.3 LonTalk 与 OSI 的七层协议比较表

	OSI 层次		标准服务	LON 提供的服务	处理器
7	应用层		网络应用	标准网络变量类型	应用处理器
6	表示层		数据表示	网络变量，外部帧传送	网络处理器
5	会话层		远程遥控动作	请求/响应，认证，网络管理	网络处理器
4	传输层		端对端的可靠传输	应答，非应答，点对点，广播，认证等	网络处理器
3	网络层		传输分组	地址，路由	网络处理器
2	数据链路层	链路层	帧结构	帧结构，数据解码，CRC 错误检查	MAC 处理器
		MAC 子层	介质访问	P-预测 CSMA，碰撞规避，优先级，碰撞检测	
1	物理层		电路连接	介质，电气接口	MAC 处理器，XCVR

(2) LonTalk 协议的报文服务

LonTalk 协议提供了四种类型的报文服务，这些报文服务除请求/响应在会话层实现外，其他三种都在传输层实现。

- 应答方式（Acknowledge），或者是端对端（End To End）的应答服务。每一个接收到报文的节点都分别向发送方应答，如果发送方在应答时间内没有全部收到应答，将重新发送该报文，重发次数和应答时间都是可选的。报文应答服务是由网络处理器完成的，不必由应用程序来干预。报文传输号确保节点不会收到重复的报文。
- 请求/响应方式（Request/Response）。每一个接收到报文的节点都分别响应发送方，如果发送方在响应时间内没有全部收到响应，发送方将重新发送该报文，重发次数和响应时间都是可选的。报文响应服务可以包含数据，是由应用处理器完成的，适合远程过程调用和客户/服务器方式的应用。
- 非应答重发方式（Unacknowledged Repeated）。当一个节点发送报文到另一个节点或一个分组时，不需要每一个接收到报文的节点向发送方应答或响应，而采用重复多次发送同一报文，使报文尽量可靠地被接收方收到。这种方式适合于节点较多的分组的广播发送，从而避免因节点响应或应答而使网络过载。
- 非应答方式（Unacknowledged）。当一个节点发送报文到另一个节点或一个分组时，不需要每一个接收到报文的节点向发送方应答或响应，也不必重复多次发送同一报文，只发一次即可。这种方式适合对可靠性要求不高，但需要速度较快，长度较长的报文。

(3) LonTalk 协议的网络认证

LonTalk 协议支持报文认证，收发双方在网络安装时约定一个 6 字节的认证字，接收方判断后只接收经过发送方认证的报文。

9.4.3 Profibus

Profibus（Process Fieldbus）是作为德国国家标准 DIN 19245 和欧洲标准 EN 50170 的现场总线，其应用范围如图9.22所示。

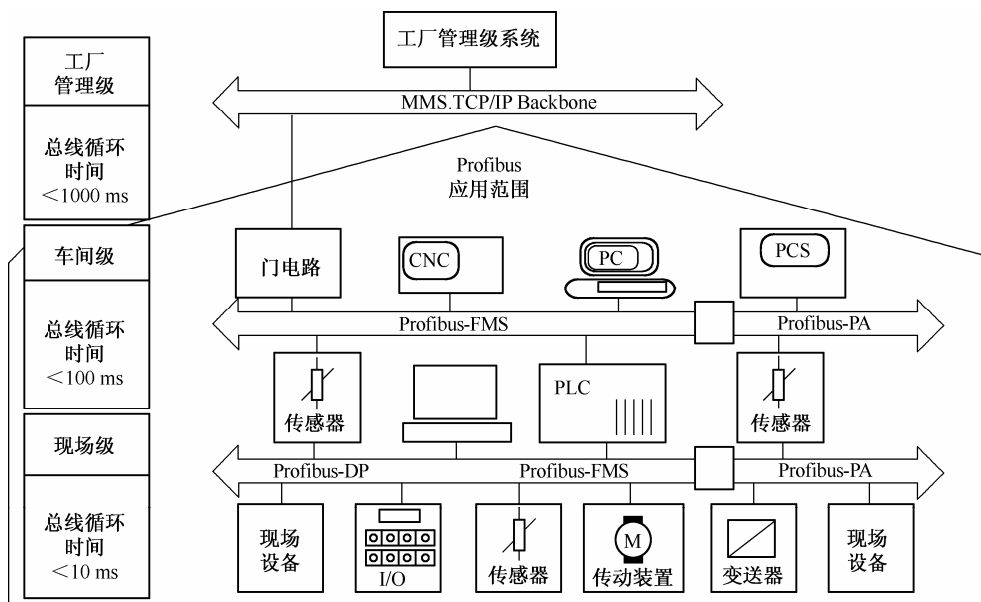


图 9.22 Profibus 应用范围

1. 组成系列

Profibus 由 Profibus-DP, Profibus-FMS, Profibus-PA 组成系列。

(1) Profibus-DP

经过优化的高速廉价的通信连接，专为自动控制系统和设备级分散 I/O 之间提供高速数据传输，可取代 24 V 或 0~20 mA 并行信号线。

(2) Profibus-FMS

解决车间级通用性通信任务，提供大量的通信服务，完成中等传输速度的循环和非循环通信任务，用于纺织工业、楼宇自动化、电气传动、传感器和执行器、可编程序控制器、低压开关设备等一般自动化控制。

(3) Profibus-PA

专为过程自动化设计，标准的本质安全的传输技术，实现了 IEC 1158-2 中规定的通信规程，用于对安全性要求高的场合，以及由总线供电的站点。

2. 协议结构

Profibus 也以开放系统互联网络 OSI 为参考模型，结构如图9.23所示。

Profibus-DP 使用第 1, 2 层和用户接口，第 3 层到第 7 层未加以描述，这种流体型结构确保了数据传输的快速和有效。

Profibus-FMS 对第 1, 2, 7 层均加以定义，应用层包括现场总线信息规范 (Fieldbus Message Specification, FMS) 和底层接口 (Lower Layer Interface, LLI)。LLI 协调了不同的通信关系，并将 FMS 的服务映射到第 2 层的现场总线链路控制 (FLC) 服务。

Profibus-PA 数据传输采用扩展的“Profibus-DP”协议和描述现场设备行为的行规，确保其本质安全性并使现场设备通过总线供电。

Profibus-DP 和 Profibus-FMS 系统使用了同样的传输技术和统一的总线访问协议，因而这两套系统可在同一根电缆上同时操作。

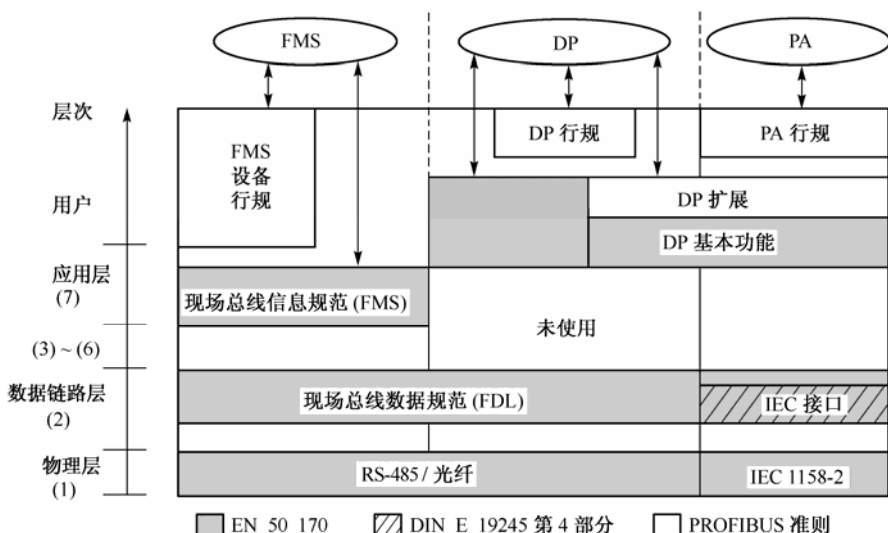


图 9.23 Profibus 协议结构

3. 传输技术

Profibus 传输有以下三种类型：

(1) DP 和 FMS 的 RS-485 传输

这是 Profibus 最常用的一种，采用屏蔽双绞铜线电缆，适用于需要高速传输和设施简单便宜的各个领域。总线结构允许增加或减少站点，分步投入不会影响到其他站点的操作。每段 32 个站点，带转发器可多至 127 个。传输速率可选用 9.6 kbps~12 Mbps，电缆的最大长度相应为 1200~100 m 每段。

(2) 用于 PA 的 IEC 1158-2 传输技术

IEC 1158-2 传输技术是一种位同步协议，可进行无电流的连续传输，能满足化工和石化工业的要求，保持本质安全性并使现场设备通过总线供电，主要特点是：

- 每段只有一个电源，站发送信息时不向总线供电；
- 每站现场设备所消耗的为常量稳态基本电流，现场设备的作用如无源的电流吸收装置；
- 主总线两端起无源终端线的作用，设计时可采用冗余的总线段，用以提高可靠性；
- 允许使用线形、树形和星形网络；
- 允许外接电源设备根据 EN50170 标准，带有适当的隔离装置将总线供电设备与外接电源设备连在本质安全总线上。

(3) 光纤传输技术

光纤可以增长高速传输的最大距离，50 m 以内可使用便宜的塑料纤维导体，专用的总线插头可将 RS-485 信号与光纤信号相互转换，使用在同一系统上。

4. 总线存取协议

Profibus 可使分散式数字化控制器从现场底层到车间级网络化，支持主-从系统、多主站系统、多主多从混合系统等几种传输方式。可采取点对点通信、对所有站点广播（不要求应答）、有选择地向一组站点广播。

系统中，主站决定总线的数据通信，得到总线控制权（令牌）的主站没有外界请求也可以主动发送信息；从站为外围设备，没有总线控制权，仅对接收到的信息给予确认或当主站发出请求时向它发送信息。

Profibus 采用如图9.24所示的混合访问方法：在 Profibus 主设备之间采用令牌。令牌只在各主站之间通信时使用，令牌传递程序保证了每个主站在确切的时间框内得到总线存取权（令牌），令牌在所有主站中循环一周的最长时间是事先规定的。

从设备由主设备循环查询，得到令牌的主站可在一定时间内依照主从关系表与所有从站通信，也可依照主主关系表与所有主站通信。

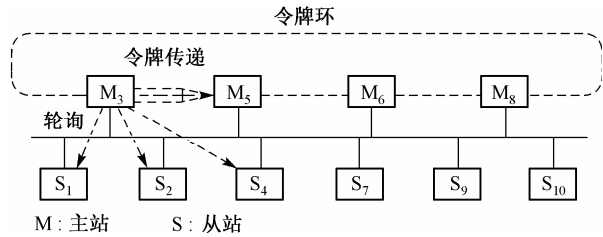


图 9.24 Profibus 总线存取协议

9.4.4 CAN

CAN 是控制器局域网络（Controller Area Network, CAN）的简称，最早由德国 BOSCH 公司为汽车内部测量与执行部件之间的数据通信而设计。由于 CAN 总线的数据通信具有突出的可靠性、实时性和灵活性，其应用范围已不再局限于汽车行业，而向过程工业、机械工业、纺织机械、农用机械、机器人、数控机床、医疗器械及传感器等领域发展。CAN 已经形成国际标准，并已被公认为几种最有前途的现场总线之一。

其特点可概括如下。

1. 协议模型

CAN 协议也是建立在国际标准组织的 OSI 模型基础上的，不过，其模型结构只有三层，即只取 OSI 底层的物理层、数据链路层和最上层的应用层。

2. 多主方式工作

网络上任一节点均可在任意时刻主动地向网络上其他节点发送信息，不分主从，通信方式灵活，无须站地址等节点信息。利用这一特点可方便地构成多机备份系统。

网络上的节点信息分成不同的优先级，可满足不同的实时要求，高优先级的数据最多可在 134 μs 内得到传输。

3. 采用非破坏性总线仲裁技术

当多个节点同时向总线发送信息时，优先级较低的节点会主动地退出发送，而最高优先级的节点可不受影响地继续传输数据，从而大大节省了总线冲突仲裁时间。尤其是在网络负载很重的情况下，也不会出现网络瘫痪情况（以太网则可能）。

4. 适应性强

只需通过报文滤波即可实现点对点、一点对多点，以及全局广播等几种方式传送接收数据，直接通信距离最远可达 10 km（传输速率 5 kbps 以下），传输速率最高可达 1 Mbps（此时通信距离最长为 40 m），节点数主要取决于总线驱动电路，目前可达 110 个，报文标志符可达 2032 种（CAN2.0A），而扩展标准（CAN2.0B）的报文标志符几乎不受限制。

5. 抗干扰能力强

采用短帧结构，传输时间短，受干扰概率低，检错效果好。每帧信息都有 CRC 校验及其他检错措施，保证了极低的数据出错率。当节点严重错误时，具有自动关闭的功能，以切断该节点与总线的联系，使总线上的其他节点及其通信不受影响。

6. 通信介质选择灵活，可为双绞线、同轴电缆或光纤

1993 年 ISO/TC22/SC3（公路车辆技术委员会电气电子分委员会）发布了用于公路车辆数字信息交换的高速通信 CAN 标准 ISO 11898 和低速通信 CAN 标准 ISO 11519。

与之相关的还有两个低速的物理接口标准 ISO 11992 1 和 ISO 11898 3。

ISO 11992 1 提高了发送电平，具有很高的抗电磁干扰能力，能抵御 100 V 以上的共模射频电压和约 6 V 的地电位差，可用于严酷的电磁环境。

ISO 11898 3 的特点是具有总线管理功能和容错性能。当总线 CAN H 和 CAN L 出现某一根断线，对电源短路或对地短路，相互短路及线路与终端电阻断开时，均能给出指示。总线在正常状态时为差分传输，故障时转为单线传输，故障消除后又自动转为差分传输。

由于 CAN 没有规定应用层和物理接口，一些组织给它制定了不同的应用层和物理接口标准，构成了几种完整的现场总线协议，其中比较著名的有 DeviceNet，SDS 及 CANopen 等。

9.4.5 现场总线的困境

1. 标准不一阻碍了现场总线的发展

现场总线技术的出现和成熟，促使工业控制系统由集散控制系统向现场总线控制系统过渡。现场总线控制系统把各个分散的测量、控制设备变成网络节点，以现场总线为纽带，把它们连接成可以相互沟通信息，共同完成控制任务的网络控制系统，真正实现了“信息集中、控制分散”的全分布式系统。但从目前看来，制约现场总线发展的关键因素是多种标准不统一的现场总线并存，难以沿开放的方向发展。

据不完全统计，世界上各大控制厂商及标准化组织曾推出过近 200 种互不兼容的现场总线协议标准，较具影响的有 40 余种。前述几种典型的现场总线，技术较成熟，市场占有率大，但这些总线之间的互联互通是当前存在的大问题。

标准意味着市场和利益。为了使自己的技术与产品占领国际市场，围绕着现场总线的国际标准展开了技术上、经济上与政治上互不相让的激烈斗争。IEC TC65（负责工业测量和控制的第 65 标准化技术委员会）于 1999 年底通过了 8 种类型的现场总线作为 IEC 61158 国际标准，但是经过 9 次投票未能通过，协商、妥协的结果，最终不得已包含了多种互不兼容的协议。

除了前述 FF 总线的 H1 和 H2、Profibus 外，IEC 61158 还包括：

Control Net International 组织支持的 ControlNet 监控级总线，它的底层（设备级）总线为 DeviceNet，两者有着共同的应用层。

欧洲现场总线标准的另两项：P Net，主要用于食品、饲养业、农业及工业一般自动化；World FIP，主要用于过程控制和制造业的现场级。

SHIP STAR 协会的 SwiftNet，主要用于航空航天领域。

Interbus Club 的 Interbus，主要用于制造业的现场级（设备级）或一般自动化。

IEC 及 ISO 还制定了一些特殊行业的现场总线国际标准，例如，基于 CAN（控制器局域网）总线的 ISO 11898 和 ISO 11519；IEC TC9（铁路电气设备技术委员会）发布的国际标准 IEC 61375 列车通信网（TCN），该标准定义了两层总线：列车总线（Wire Train Bus，WTB）和多功能车辆总线（Multifunction Vehicle Bus，MVB）；IEC SC17B（低压配电与控制装置分委员会）发布的国际标准 IEC 62026 低压配电与控制装置—控制器与设备接口；IEC TC44（机械设备电气安全技术委员会）1995 年 11 月发布的 IEC 61491 工业机械电气设备—控制单元与驱动装置之间的实时通信串行数据链路。

此外，还有一些总线也获得了广泛应用。例如，美国的列车通信标准也采用了 LonWorks；英国的 ERA，挪威的 FINT，以及日本三菱的 CC LINK，施耐德公司的 Modbus 等。

2. 现场总线开放性的障碍

开放性应该是现场总线的主要特点之一，但现有国际标准的异种现场总线之间并不能进行互操作，虽说可以通过网关（Gateway）通信或通过 OPC（OLE for Process Control）协议进行互操作，但这种互操作只能在各自的主机间进行，不能在彼此的现场仪表间进行，缺乏控制的实时性，因而没有实际意义。

标准不一，种类过多，严重制约了现场总线控制系统的发展。IEC 61158 并没有实现统一总线标准的初衷，不能使设备制造厂商有所遵循，也使用户难以选择。

实际应用中，一个系统很可能采用多种形式的现场总线，如何在实际中根据应用对象，将不同层次的现场总线组合使用，使系统的各部分都选择最合适的现场总线；如何把工业控制网络与数据网络进行无缝的集成，从而使整个系统实现管控一体化；组态时如何选取功能块；设计网络布局时，如何考虑各现场节点的距离、节点之间的功能关系，如何使网络上的信息流动最小化；如何进行通信参数的组态，在系统的实时性与网络效率之间做好平衡……，对用户来说，系统的集成仍然是比较困难的问题。

3. 现场总线的技术瓶颈

(1) 从系统安全考虑，希望即使出现总线电缆截断等严重故障，也只是降低系统的效能，而不至于导致整个系统的崩溃，但目前许多现场总线不能保证这一点。

(2) 本质安全防爆关于总线的长度和总线上负载数量的规定，限制了现场总线节省线缆优点的发挥。

(3) 组态参数的设定对系统性能影响很大，但不少现场总线系统组态参数过多而且复杂，不易掌握。

9.4.6 工业以太网

1. 工业以太网的优势

有条件、不彻底的开放性阻碍了现场总线的发展，但具有传输速率高、低耗、易于安装和兼容性好、支持几乎所有流行网络协议的以太网（Ethernet）技术却得到了迅猛发展，成为目前世界上应用最广的局域网技术。TCP/IP 通信协议已是国际通用的标准，TCP/IP 协议极其灵活，几乎所有的网络底层技术都可用于传输 TCP/IP 的通信。开放的、透明的、应用 TCP/IP 协议的以太网已成为最流行的分组交换局域网技术，不仅在办公自动化和各个企业的管理网络得到了广泛使用，并且在工控领域通信这一关键环节具有无可比拟的优势（见表 9.4），适应了工业自动化系统向分布化、智能化控制方面发展的需要，在工业控制局域网中得到了很好的应用。

表 9.4 工业以太网的优势

	工业现场总线	工业以太网
故障隔离	一般无故障隔离功能（如信号短路问题）	有故障隔离功能
故障检测	难于确定故障源，不便于维修	有网管，故障定位准确
开放性	部分厂家联盟	支持各家产品的数据传输
数据带宽共享	不能	不同协议可以共享同一链路
硬件链路共享	需要重新设计网络系统和布线	不同协议可以共享同一链路
设备更新提升	需要重新设计网络系统和布线	可以在原有的网络上扩展
网络速度	最高 12 Mbps，不同网段速度固定	根据需要 10～1000 Mbps 以上
物理媒介	主要是双绞线，光纤	双绞线，同轴电缆，光纤，无线可以混合成网
互连性	难以互连	易互连，并直接与 Internet 接合，与管理网融合
硬件互换性	不同厂家的产品不一定能连接和替代	不同厂家的产品可以替代和连接

(1) 以太网和 TCP/IP 进入现场总线控制系统的底层工业现场,使得工厂的管理可以深入到控制现场,也就是说,可以将以太网建到现场 I/O 级,从而实现办公自动化与工业自动化的无缝结合,构成“从会议室到传感器”,企业管控一体化的企业信息网,提升企业核心竞争力。

(2) 在工业控制领域,随着仪器仪表智能化的提高和工业管理自动化的深入,系统传输的信息将不满足于几十字节的数据,而是大量的多媒体信息,以太网可以满足工业控制网络对大容量传输的需求。

(3) 具有 TCP/IP 接口的现场设备可以无须通过现场的计算机,直接连接 Internet,实现远程监控或远程维修。

(4) 底层网络引入以太网不仅使现场层、控制层和管理层在垂直层面方便集成,更能降低不同厂家设备在水平层面上的集成成本,解决了工业底层网络的现场总线协议数目众多,不同协议无法兼容的困扰。形成真正意义上的全开放、可互操作的现场总线控制系统。

2. 以太网技术的改进

当初为办公自动化设计的以太网,并没有考虑工业要求,缺乏确定性、鲁棒性差,存在不可预见性,在车间底层网中应用是脆弱的。

这是由以太网本身的 CSMA/CD 协议造成的,初期以太网速度较低,如果当网络比较忙碌,总线竞争多,或冲突节点相距甚远时,可能发生某些节点的信息不能及时发送,甚至长时间发不出去的情况,这在工业控制中是不允许的。

以太网技术的改进已经使这种局面得到了根本的改观。

(1) 通信速率的提高

以太网速度每隔 3~4 年就提高一个数量级,目前千兆以太网已在局域网、城域网中普遍应用,万兆以太网已在 2003 年确定了国际标准,由“多模光缆布线系统”加“拥有 10 Gbps 模块的网络交换机系统”组成的完整解决方案也已进入实用。对于同样的通信量,通信速率的提高意味着网络负荷的减轻,也就是确定性的提高。在网络负荷不超过 36% 的情况下,以太网发生碰撞的可能性极小。

(2) 交换技术的快速发展

快速交换式以太网技术,采用全双工通信,可以完全避免 CSMA/CD 中的碰撞,并且可以方便地实现优先级机制,保证网络带宽的最大利用率和最好的实时性能,完全避免了 CSMA/CD、主从、令牌等可能的低效率。交换式以太网技术已经消除了以太网应用于控制领域的障碍。交换器把网络分割成互不相连的几个网段,每个设备独占一个网段,从而大大降低了冲突的可能性。使得多个网上设备之间同时进行通信时不会有冲突发生。以太网核心交换机的交换速度可达 40 Gbps,直接传输距离可达 70 km。

(3) 网络产品的工业级标准

在当前竞争激烈的经济环境中,工业网络向开放、透明系统方向发展的趋势,使 PC 控制能方便地接入 Intranet 或 Internet,其中最为重要的两个标准是 TCP/IP 通信协议和以太网网络架构。

虽然工业自动化领域的以太网标准与办公自动化领域中的以太网标准相同,但二者对网络产品的要求是完全不同的。通常,工业自动化网络故障引起的生产过程中断代价昂贵,需要能在强电磁干扰、高温等极端条件下正常工作,对可靠和容错的要求很高。工业级以太网交换机提供了数据处理的高速度和容错保护,并且无须其他特殊硬件支持,其灵活的扩展功能为用户提供了网络柔性。

在数据传输链路方面，由于骨干网的交换机都带有 SNMP（简单的网络管理协议）管理单元，集成了基于 Web 的管理系统，支持 RMON 网管，从而保证了良好的安全性。对交换机的管理采用了二级密码访问机制，可以关闭不用的交换机端口，防止外部设备的非法接入。千兆以太网交换机支持工作组管理，可以根据需要将端口划分为不同的工作组，并限制其访问权限，具有严格的安全保证。

工业级以太网产品，采用了高容量大负载的工业级设计，配置双电源，采用无源背板，大大提高了系统的可靠性。其工作环境温度范围可以达到 0~50℃，防护等级达 IP30；支持“超级环”（HIPER-RING）、双链路冗余（Link Redundancy）、双节点冗余（Dual-homing）和生成树（Spanning Tree）等多种冗余方式。可以构成冗余光纤工业以太环网，当发生某处光纤故障时，网络通信会在 500 ms 之内恢复正常。

这些改进都为以太网进入工业控制领域铺平了道路，现在已没什么理由再说以太网不能建立一个高效、开放、有确定性的现场总线系统。

3. 嵌入式以太网的发展

利用嵌入式软硬件，设计者可以在单片机系统上实现以太网技术即嵌入式以太网。嵌入式以太网的出现为现场总线控制系统的设计者提供了现场设备通信的新途径。

在硬件方面，目前国外一些著名大公司已经利用嵌入式技术将以太网用于它们最新推出的现场总线控制系统。如 GE 公司将以太网接口做在保护装置中，GE HARRIS 公司推出了带双以太网接口的 D25 测控单元装置，ABB 也推出了带以太网接口的间隔控制器 R580。

在软件方面，广泛用于各种嵌入式智能设备的嵌入式操作系统是美国集成系统公司推出的 pSOSystem，简称 pSOS。

pSOS 采用了先进的模块化体系结构，包括一个实时多任务核心和一系列的公司软件部件和连接库。系统中的每个部分都是封闭式的，相互之间既独立又密切协作，这在很大程度上提高了 pSOS 的可扩展性。开发人员可以根据不同的应用需求来制定操作系统的功能和所需要内存的大小。这种高度的可扩展性还使简单的独立系统能够轻松地升级到复杂的网络化多处理器系统。目前，pSOS 已经能够支持范围极其广泛的各种硬件产品。

另外，Windows CE 也不失为一个嵌入式操作系统较好的选择，特别是其新版本的推出，在确定性、实时性、优先级扩展能力及中断服务延时、进程切换时间上有了显著的提高，过程处理循环时间降到毫秒级。事实上，一些公司已生产出采用 WinCE 的控制装置，如 WinPLC，GE 公司的 CNC，Xycom 自动化公司的 ASIC 3000 等。

4. 高速以太网（High Speed Ethernet, HSE）的发展

在以太网进军工控领域的过程中，各大现场总线组织都予以高度重视，并纷纷出台了自己的发展计划。例如，现场总线基金会在 2000 年 3 月颁布了 HSE 最终技术说明书。

HSE 支持基金会现场总线（FF）协议的各个方面，比如，功能块和设备描述等，而且能充分利用已经成熟的以太网技术。

HSE 使用以太网现成的交换机、路由器、网线及光纤，连接成各种拓扑和规模的网络。HSE 结构应用标准以太网协议，如 TCP/IP，SNTP，SNMP 等，并专为多重 H1 现场总线信息在主干网上高速高效的传输而设计。它还应用 FF 标准功能块如 AI，AO 和 PID 等，保证控制网络上各层数据表达的一致性。

HSE 技术的一大特点是连接器（linking device）。HSE 连接器融合网桥和网关功能。连接器用于连接 H1 现场总线各个网段，并提供对 HSE 主干网的接口，其速度为 100 Mbps 或 1 Gbps。

它的网桥功能使 H1 设备通过连接器传输接口相互连接，而不用上层系统的干预。连接在连接器上的 H1 设备可以通过 HSE 主干网与连接在另一连接器上的 H1 设备通信，而 HSE 主机可以与所有连接器和设备通信。这项功能可以使关键数据提供给远方的现场设备用于监控及报表。

连接器的网关功能可以使 HSE 网络与其他底层控制或信息网络相连。HSE 连接器不需要对 H1 子系统进行信息解释。它的功能是集合 H1 现场总线网段的信息，并将 H1 地址转化成 Internet 的 IP 地址。对其他网络，监控和控制参数可以简单地设定到标准 FF 功能块或新的“灵活功能块”。

灵活功能块（FFB）是 HSE 技术又一独特的功能。FFB 用于先进过程离散控制，并集成远程 I/O 设备和子网。虽然它是 HSE 的一部分，但也能应用于 H1 系统，它包括多入多出模拟器、离散 FFB 和为特殊控制算法设计的专用 FFB。

专用 FFB 用符合 IEC 61131 标准的编程语言编写。支持如获得管理数据、批处理和 I/O 接口界面等功能。

FFB 的使用提高了控制功能下放的能力，而它是 FF 独特和重要的特点。HSE 连接器和现场设备组成在现场层的简单系统，拥有批处理和逻辑控制功能。从而在许多应用中，可以不需要上层控制器，降低设备成本和空间消耗。

将控制功能集成在 HSE 连接器中可以使控制结构从控制室延伸到现场。连接器使用于通信点密集的地方，如阀门和执行器等。把控制功能通过连接器分散到现场还可以减少集中控制中的错误和遗漏。

HSE 还支持本质安全。在危险的作业区域，通过光纤将以太网与其中的连接器相连。利用连接器的控制功能，可以减少现场 I/O 设备和控制器的数量。

现场总线基金会明确表示 HSE 就是一种具有控制功能的 Internet。HSE 连接器将远端 H1 网段的信息传递到以太网主干线上。这些信息通过 Internet 传到主控室，操作员通过网络浏览器就可以监控生产过程，控制信息也可同样被管理系统获取。

将 Internet 与高效的以太网结构结合起来，HSE 可以成为企业理想的把控制与管理联系起来的现场总线网络。

5. 工业以太网的应用实例

实际应用说明，以太网技术的发展使它有能力和进军工业领域，也可以适应恶劣的工业环境。例如：

(1) 北海石油钻井平台的灌浆系统应用了工业以太网并且工作良好，系统中的以太网控制器和 I/O 端口通过了包括冲击、振动，以及各种环境条件（如潮湿，低温等）在内的 IEC 的 14 项工业标准测试。实际运行表明，以太网能够适应恶劣的工业环境。

(2) 澳大利亚一个金矿的控制系统采用工业以太网作为主干网，构成了包括 50 台计算机，148 台可编程逻辑控制器和 420 000 个 I/O 端口，能监视 63 000 个监测点的庞大控制系统。它的主机系统软件采用 Windows NT。实际运行表明，以太网不仅能胜任实时控制和大数据量传输，而且可能是唯一能适应工业大系统需求的网络。

6. 工业以太网的前景展望

开放、透明的标准才是最有生命力的标准。随着以太网的实时性、稳定性和抗干扰性问题的逐步解决，工业以太网成为目前最活跃、性价比最高、最有发展前景的现场总线技术之一，大规模进入工业控制领域将指日可待。以太网应用到工业控制领域，可以形成真正意义

上的开放式系统，实现不同厂家的以太网产品的互连。如果说现场总线系统是工业控制领域里的一场革命，那么以太网技术将把现场总线系统带入又一个新时代。以太网应用于工业现场设备层是网络发展的趋势，也是信息技术发展的必然结果。

(1) 优点

由于嵌入技术使以太网能到达传感器和执行器，解决了整个网络的瓶颈效应集中在应用现场总线的设备层上的问题，新型现场总线控制系统的特点是：

① 以太网贯穿于整个网络系统的各个层次，使网络成为透明的覆盖整个企业范围的应用实体。实现了办公自动化与工业自动化的无缝结合，其良好的互连性和可扩展性使之成为一种真正意义上的全开放、大统一的网络体系结构。

② 采用以太网作为现场总线，保证现场总线技术的可持续发展。普遍应用的以太网，一直受到广泛的重视和大量的技术投入，保证了以太网技术的不断发展。为工业控制领域采用以太网作为现场总线，提供了技术上可持续发展的强大后盾，并在技术升级方面无须独自的研究投入，这是任何现有的现场总线技术无法比拟的。

③ 低成本是基于以太网的现场总线系统的最大优越性。采用 TCP/IP Ethernet 架构，则无论是 Cable、Connector、Switching、Hub、网络接口乃至软件开发环境，皆与主流市场相同，很容易达到资源共享。并且随着集成电路技术的发展，其价格还会进一步下降。由于以太网已被使用多年，积累了丰富的设计、应用经验，系统的开发和培训费用可以显著降低。

(2) 缺点

① 不同的应用场合有不同的特点，一种现场总线（以太网）难以满足所有要求。例如，有的场合需要高抗干扰；有的场合需要远距离传输，但并不需要大的带宽；有的场合不适宜采用星形网络，等等。

② 采用金属导线的以太网，单个网段长度有限，光纤网成本较高且不能实现总线供电。

③ 采用金属导线的高速网络的抗干扰能力比低速网络差。

④ 在简单的现场设备尤其是开关量现场设备的应用中，以太网在成本和性能方面不占优势。

⑤ 即使大家都使用工业以太网和 TCP/IP 协议，但商业竞争的结果还可能导致使用不同的应用层协议。

9.4.7 信号线与电源线合用电缆的研究

在使信号线尽量远离干扰源的同时，在信号线与电源线合用电缆方面也进行了大量的研究，多种现场总线可以通过总线向现场设备供电，例如，FF 总线、Profibus PA 的 IEC 1158-2 传输技术、LonWorks 现场总线的电源线收发器技术；LonWorks 电力线收发器还可以将通信数据调制成载波信号或扩频信号，通过耦合器耦合到 220 V 或其他交直流电力线上进行数据通信。

近年来进展比较突出的还有：

1. PLC（Power Line Communication）技术，俗称“电力线上网”

利用电力线传输数据和话音信号的通信方式最早出现于 20 世纪 20 年代初期，用做电力网长距离调度的通信手段。电力线通信的难点在于电器的启停、运行都会产生较大的间歇性噪声，信号衰减很快，线路阻抗也经常波动。近年来，随着 Internet 技术的飞速发展，利用 220 V 低压电力线传输高速数据的价值越来越为人们所重视，因为它利用既存电力线而不用另行布线、覆盖范围广、连接方便，被认为是提供“最后一千米”解决方案中最具竞争力的技术之一。

由于电力线上网的潜在市场空间，包括我国在内世界许多国家纷纷投入研究和开发。已有多家公司成功地进行了采用传统的输电线路进行 Internet 接入的试验。

2. 利用电力线载波进行智能家居控制

智能家居系统可以对各种照明、家用电器和用电设备进行集中控制、电话远程控制、可编程定时控制及遥控。不需另行布线，利用插在家居中任意一个电源插座上的控制器，就可以控制任意一个（组）灯或电器，也可以控制整个家居用电网络，实现资源共享，其控制对象可以任意设置和改变，实现全开全关，场景设置，多控开关等复杂操作功能。

其中，较有代表性的 X10 电力载波通信协议传输数据的规定如图9.25所示：

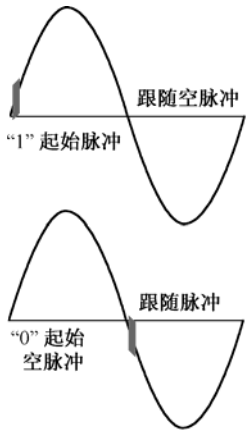


图 9.25 X10 协议传输数据的规定

在交流电过零点后，发送 1 ms 的 120 kHz 脉冲（60 Hz 电力系统）；二进制数“1”定义为前一个过零点有脉冲，下一个过零点无脉冲；“0”则相反。

过零点至今仍然是最干净、最不易受干扰的地方。1-0、0-1 补码方式也提高了信号传输的可靠性。

3. 通过以太网数据传输线传送小功率电源

2003 年通过的 IEEE 802.3af 标准允许以太网在数据传输线上同时传送直流电源，其功率约为 13 W，小型数据设备可直接用以太网连接来供电。

在以太网系统中，提供电源的设备（Power Sourcing Equipment, PSE）通常是以太网交换机、路由器、集线器或其他网络交换设备，使用电源的设备（Power Device, PD）

则有多种形式，如数字 IP 电话、无线网络接入点、PDA 或笔记本电脑接驳器、移动电话充电器及 HVAC 温度自动调节器，等等。实际上，任何需要数据连接并能在 13 W 或更低功率下工作的设备从 RJ45 插座就能够得到相应的电力。

PSE 要求 CAT-5 以太网电缆中信号线对或备用线对（但不是两者同时）提供额定 48 V 直流电源，电源作为两个加电的线对间共模电压差，加到差分数据信号与线缆耦合的隔离变压器中间接头，发送线对和接收线对之间的 48 V 电位差不会对两端数据收发器产生影响。

习题与思考题

- 1 哪一种总线操作称为寻址？通常寻址方式有哪几种？应该怎样选择？
- 2 微机控制系统有哪些数字信号传输方式？它们各有什么优点和缺点？
- 3 令牌传递总线中，等待时间的上限和下限各是多少？为什么？
- 4 网络拓扑结构、信号方式、访问控制方式和传输介质等主要因素对微机控制网络的性能有什么影响？
- 5 为什么只有按照 OSI 标准生产的网络产品才能实现开放系统互连？
- 6 分布式控制系统的体系结构和设计原则有什么优点？设计时要注意哪些问题？
- 7 现场总线与一般的计算机数据通信网有哪些不同？为什么说现场总线控制系统是新一代控制系统？
- 8 选择一种你比较熟悉的现场总线，简述其协议的主要内容并分析其优缺点。
- 9 现场总线的发展遇到了哪些障碍？试提出解决办法。
- 10 工业以太网有哪些优势？在今后的发展中可能会出现什么问题？你认为应该怎样解决？

第 10 章 微机控制系统设计

10.1 微机控制系统设计与实现

10.1.1 微机控制系统设计原则

微机控制的对象多种多样，系统设计的方案和技术指标也千变万化，但系统设计遵循的原则是一致的，包括安全性、可靠性、实时性、可操作性与可维护性、通用性和经济性。

1. 安全性

安全运行是任何设备和系统存在的基础。微机控制系统的故障可能导致严重后果，在设计初始阶段就应当将安全性作为最基本的技术要求。无论是硬件的设计，还是软件的设计，都应当考虑故障安全问题：一旦系统发生故障，不至于使系统发生可能产生危及安全的后果。例如，列车控制系统的故障只允许导致停车，而绝不能引起列车的颠覆。

2. 可靠性

高可靠性是微机控制系统设计的技术前提。设计的第一步是选用高可靠性的计算机系统、可靠的外围电路、有效的信号处理装置，以及特殊的加固措施等。一些特殊的控制系统可考虑采用双机冗余的方式。在系统设计的初期，就应当明确设备的运行环境，这些环境对系统可靠性的影响，以及对设备可靠性的要求和应当采取的技术措施。

3. 实时性

实时性，包括时间驱动方式和事件驱动方式，是微机控制系统实现正常检测与控制的基础。输入通道应当输入包含有效信息成分的采样信号。输出通道同样应当在规定的时间内或事件允许的周期内，输出有效的控制指令。应当根据可靠性和实时性要求，设计合理的结构方案，提出相应的硬件技术指标和软件应当达到的技术要求等。

4. 可操作性与可维护性

操作方便表现在简单、直观、形象、便于掌握。任何新技术的使用，其前提是要为人类服务，而不是给使用者带来不便，甚至引起使用者的拒绝。

维修方便体现在易于查找故障，易于排除故障。采用标准的功能模板结构和热插拔方式，便于更换故障器件或模板。

5. 通用性

设计时应考虑能适应不同的控制要求，采用积木式结构能灵活地构成系统，并能灵活地进行扩充。

通用性体现在两方面：硬件模板设计采用标准总线结构，配置各种通用的功能模板；软件模块或控制算法采用标准模块结构，使用时可按要求选择各种功能模块，灵活地进行控制系统组态。

6. 经济性

技术永远是为经济服务的，微机控制技术应该带来高的经济效益，这就要求系统设计的性能价格比尽可能高，投入产出比（包括人力、物力、财力、时间和市场要求各方面）尽可能低。

10.1.2 系统的设计与实现过程

通常包括准备阶段、设计阶段、仿真调试阶段和现场安装调试阶段等。

1. 准备阶段

在设计研制准备阶段，应当对整个工程任务有充分的了解。甲方应当提出任务委托书，乙方（设计研制任务承担方）应当充分深入地研究任务委托书，然后双方根据技术条件的限制、经济与环境条件及双方对问题的理解，进行协商、修改和确认任务委托书。

确认任务书后，乙方应提出系统总体技术方案，清楚地反映出三大关键问题：技术难点、经费概算和工期。在此基础上，乙方还应当进一步完成可行性论证，包括技术可行性、经费概算和工期的进度计划。然后，在双方意见一致的前提下，签订合同，这是双方以后合作的唯一依据和凭证。

2. 设计阶段

设计阶段的第一个工作是组成一支合适的设计队伍，队伍内部应当有明确的分工和互相协调的关系。

在初步的技术总体方案的基础上，完善技术总体方案，将技术方案从粗线条到明确的任务分解和实现过程进行分解细化。技术方案包括硬件方案和软件方案及其相互关系。

总体设计应建立的技术文档内容包括：

- 系统的总体结构；
- 系统的主要功能、技术参数与指标、原理性框图及文字说明；
- 方案的比较和选择；
- 保证性能指标要求的技术措施；
- 控制策略与实现的算法；
- 系统相关设备的选型；
- 硬件的结构、配置、原理图、PCB图、关键器件的参数与采购要求、调试与测试方法等；
- 软件的结构、功能、框图、编程说明、调试与测试说明、程序清单等；
- 抗干扰与可靠性设计技术；
- 机箱与机柜的设计要求和结构设计；
- 电源与供电设计、布线技术与要求等；
- 经费和进度的安排；
- 方案的论证与评审，关键技术方案的专家论证。

3. 采购、研制与外加工阶段

根据设计要求和进度，完成全部软件编程、硬件制作、部件制造、相关设备技术与研制，并完成部件的仿真调试。

4. 仿真调试阶段

完成微机控制系统的设计研制后，应当进行仿真调试，以确认系统的性能、参数和功能实现。仿真调试包括硬件调试、软件调试和系统联调。实验室仿真联调应当建立仿真环境，仿真环境的建设包括数学仿真（计算机仿真）、半物理仿真（硬件闭路动态实验）和全物理仿真（模拟环境条件下的全物理仿真）。特别要指出的是，不经过系统仿真和各种实验，试图在现场调试中一举成功的想法是不切实际的。

在完成实验室仿真联调之后，还要进行考机运行。通过连续运行实验，暴露问题和解决问题。

5. 现场安装调试阶段

设计研制的最后阶段就是现场的安装、调试、试运行。根据合同要求，完成所有技术指标的测试工作，然后完成相关的技术报告和建立技术文档。在需要的时候，还应当进行振动、冲击、温度、湿度等的适应性试验，对电磁干扰有特殊要求的设备，还应当进行电磁兼容试验，所有的这些试验都应当提供相应的试验报告。最后，才能向甲方移交整个系统，包括相关的技术文档。

10.2 输入输出通道

微机控制系统必须具有各种数据采集输入和控制指令输出功能，如图 10.1 所示，其输入输出通道的信号分为模拟量的输入/输出（AI/AO）和开关量的输入/输出（DI/DO），当外部传感器的输出信号幅值很小或者与计算机的接口电平不兼容时，经过调整电路才能与计算机匹配。计算机的输出信号经过输出调整和功率驱动，才能控制外部的执行机构。调整电路也能保证信号的传输质量，防止电磁干扰和信号发生畸变。

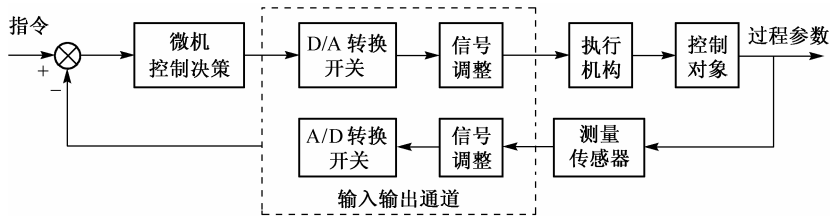


图 10.1 微机控制系统结构框图

10.2.1 开关量信号的调整

开关量（数字量）输入输出信号的调整电路（又称为 I/O 电气接口）的主要功能是，滤波、电平转换、隔离、功率驱动等。由于外部设备信号的输入可能导致瞬时高电压干扰、过电压、接口噪声、接触抖动等，因此输入信号的调整电路主要有以下电路。

1. 信号转换电路

将电压或电流输入信号转换成规定的电压信号，此时主要是选择合适的分压电阻 R_1 和 R_2 （如图 10.2 所示）；将开关触点的通断转换为规定的电平信号（如图 10.3 所示）。

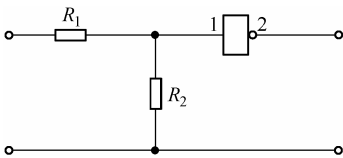


图 10.2 电压或电流转换电路

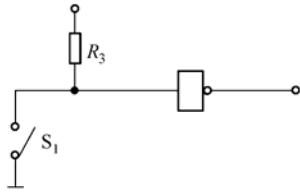


图 10.3 开关触点的电平转换电路

2. 滤波电路

微机控制系统中，大量的开关电路和长线传输、空间电磁场等都会引起噪声干扰。例如，与计算机共电源的发光二极管显示器就会影响计算机工作信号的质量，所以通常采用 RC 网络构成的无源或有源低通滤波电路。

3. 消抖电路

微机控制系统接受有触点电路如键盘、按键、开关等的输入信号时，必须解决在闭合或断开瞬间所产生的抖动问题，通常采用消抖电路或与消抖软件配合，图10.4是由RS触发器构成的双向消抖电路。

4. 吸收保护电路

为了防止超过正常情况十余倍的过电压、瞬态尖峰脉冲及反极性信号等造成损坏，常常采用由齐纳二极管 VD 或压敏电阻 U 等构成吸收保护电路（如图10.5 所示），将高幅值尖峰信号钳位在安全的电平范围内。

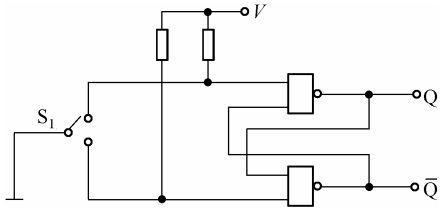


图 10.4 双向消抖电路

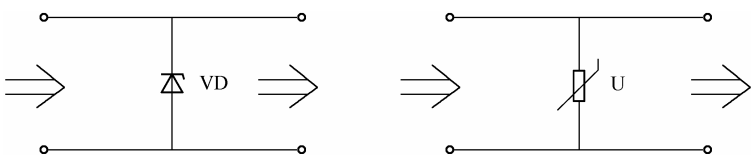


图 10.5 吸收保护电路

5. 隔离电路

光电耦合器件的可靠性高，成本低，使用方便。光电耦合电路是开关信号调整中最有效的隔离电路，只要输入输出端的电源完全隔离，光电耦合器件就可以完全隔离输入输出信号的电气关系。由于其输出端的供电电源完全限制了输出信号的幅值，因此对于尖峰干扰的吸收效果也非常好，同时还具有电平转换功能。

在实际应用时，一般都综合使用滤波电路、消抖电路、吸收保护电路和隔离电路，保证微机系统的正常工作和获取准确的外部状态采样信号。

6. 功率驱动电路

数字量输出的信号调整还包括输出信号的功率驱动，分为提高数字接口电路扇出能力的三态门输出驱动电路、开路集电极（OC 门）输出电路（如图10.6 所示）、门电路外加功率三极管组成的中小功率驱动电路、门电路加达林顿阵列的中功率驱动电路等。

工业控制领域中，大量使用数字信号输出加功率三极管驱动，外接继电器或固态继电器的功率放大方式，可以驱动各种大功率设备，如电机的启停、阀门的启闭或其他设备的动作（如图10.7 所示）。

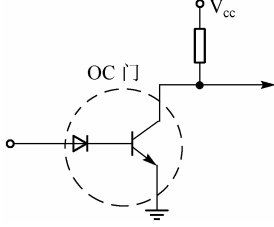


图 10.6 OC 门输出电路

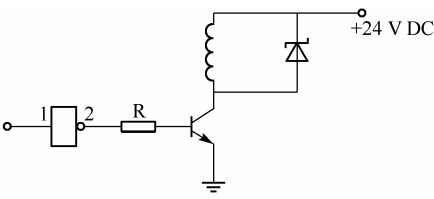


图 10.7 大功率驱动电路

10.2.2 模拟量信号的调整

模拟量信号的调整电路包括 I/V 转换和 F/V 转换。

模拟信号的输入调整还包括非电物理量的转变、电信号的形式变换与量程变换、放大、

滤波、线性化、共模抑制、环境补偿、隔离等。模拟输出信号的调整也有形式转换、放大、隔离等处理措施。

模拟量输入输出信号调整电路的产品有模板和模块两种形式，可根据具体的对象与需要选用。

常见的模拟信号调整电路包括电桥、放大、线性化、隔离、偏置等。

1. 电桥电路

普遍使用的电桥电路简单实用，通过它可以实现信号的转换，温度的补偿，变形、位移与压力等测量信号的输入，电路原理如图10.8所示。

电桥在处于平衡状态时，其输出总是等于零，而与激励的方式（输入信号）是直流还是交流，激励输入的大小，输出的检测方式是电流检测还是阻抗检测无关。如果电桥中某一个电阻发生变化，则产生不平衡输出电压差 $U_{out} = \Delta U$ （如果输入是直流输入），输出电压差与电阻的变化率成正比，而且输出信号的频率与相位和输入信号的频率与相位相同。

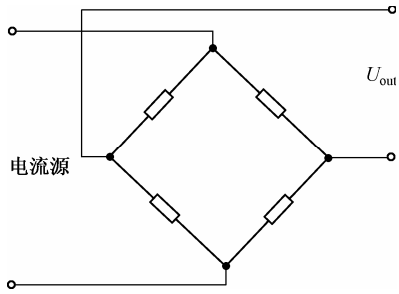


图 10.8 电桥电路基本原理

当电桥电路应用于温度检测、拉压力检测等领域时，检测量程较大，信号传输线较长，因此应当注意：

- ① 由于电阻的非线性特性和对温度的敏感性引起输出的非线性，应采取温度补偿措施，保证激励电源的稳定性及精度；
- ② 将数字处理技术及通信技术与测量传感器、电桥及前置放大电路集成，通过数据通信方式与微机控制系统相连，可以大幅度提高系统性能；
- ③ 对长距离引线导致的耦合干扰及地线环流问题，应当采取相应的抗干扰措施及合理的地线处理措施。

2. 信号放大电路

放大电路的设计与选择主要考虑精度要求（失调与失调温漂）、速度要求（带宽、上升率）、幅值要求（工作电压范围及增益），以及共模抑制要求等。

当传感器有较高的对地电压时，普通单端放大器会工作失常，应当采用共模抑制能力较好的双端差分输入放大器。如图10.9所示，由三个普通运算放大器构成的差分信号放大电路，放大线性度和共模抑制比都大为提高，其输出方程为

$$U_a = (1 + 2R_1 / R_G)(R_3 / R_2)(U_1 - U_2) \tag{10.1}$$

改变增益调节电阻 R_G 不影响差分关系。应用时必须要有输入端到地的直流通路，以免由于偏移电流对寄生电容充电导致输出漂移或失控。

3. 滤波和限幅电路

大信号的限幅电路一般采用漏电流比较小的稳压管，小信号的限幅电路可以采用漏电流小的硅二极管，但是这两种方法都会产生一定的非线性且降低灵敏度。

无源滤波一般都是采用 RC 网络。对于低频或直流等慢变化信号，电阻 $R_1 \sim R_4$ 一般取 10 kΩ 左右，电容 $C_1 \sim C_3$ 一般选 10 μF 左右的钽电容（如图10.10 所示）。

4. 共模电压与隔离

设备的绝缘不良、分布电容及静电感应等因素，特别是这些设备的电气部分对地浮空或

者对地电阻较大时，都会耦合一定的共模电压。当用电设备负荷急剧变化时，共模电压值也随之急剧变化。共模电压会通过计算机的输入输出信号传输线加到各 I/O 口上，轻则导致信号失真，影响测量精度，重则击穿器件，损伤设备。

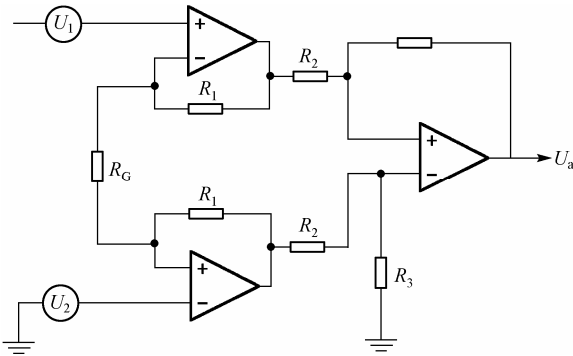


图 10.9 信号放大电路

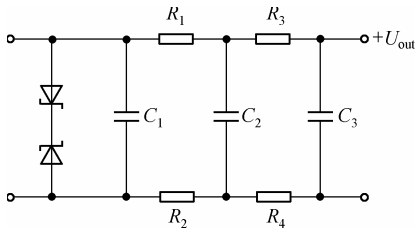


图 10.10 滤波与限幅电路

单个设备的良好接地可使共模电压消除，而对于不能接地的多个设备而言，隔离是克服共模干扰影响的有效措施。特别是当现场有多个设备互连时，隔离措施的安全可靠性更加明显。在有以下情况出现时，应采取隔离措施：

- ① 比较差分测量时若有高共模，表明差分信号源有很高的对地阻抗；
- ② 计算机与现场设备之间的距离较远，可能产生很高的地间共模；
- ③ 为了安全的原因，两个互连的设备必须分别接地，此时设备间可能产生地间共模；
- ④ 测量探头同带强电的设备直接接触或者绝缘度不高。

模拟信号的隔离通常采用将输入信号、输出信号、电源三者相互隔离的隔离放大器。隔离放大器包括光电耦合型和如图10.11所示的变压器耦合型。

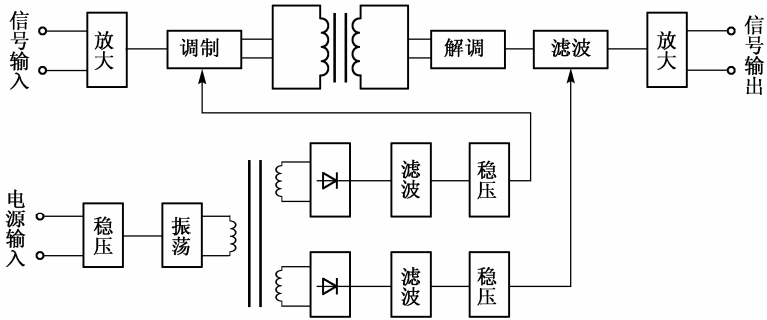


图 10.11 变压器耦合型隔离放大器

10.2.3 输入输出模板设计

数字量（开关量）和模拟量输入输出模板是实现被测控对象过程信息的输入，完成控制命令的输出的基本部件。

输入输出模板通常采用性能价格比较好的单片机，由板上的从 CPU 独立进行信息的采样输入，完成简单的数据预处理功能，如开关量的编码与解码、模拟量的数字化滤波、非线性补偿、数制转换等，然后转交给系统的主 CPU，使主 CPU 既能实时输入输出信息，又有足够

的能力处理检测控制计算任务。输入输出模板需要解决：模板上的从 CPU 的运行，申请中断并且能够被主 CPU 识别，主从 CPU 之间的通信与信息交换。

模板上的从 CPU 通过总线与系统主 CPU 之间的信息交换一般有两种方式：FIFO（先进先出）的握手联络双向并行口方式和双口 RAM 方式。

输入输出模板的可靠性与抗干扰，是设备电磁兼容的关键问题。

1. 元器件布置

微机控制系统中，I/O 模板的元器件布置通常遵循下列原则。

(1) 以功能划分区域（如图10.12 所示）

强弱电功能区域分离，高低频信号功能区域分离，I/O 信号尽量不靠近系统总线。不同的功能区的电源也应当是相互隔离的。

(2) 根据元器件位置确定连接器引脚安排

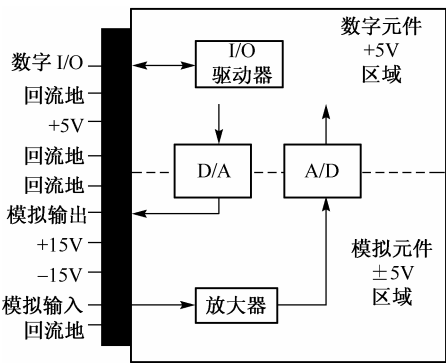


图 10.12 以功能分区的元器件布置

输入输出驱动器紧靠连接器，以防印制板上传输线过长造成耦合干扰。避免高速数字集成电路芯片靠近连接器或放置在两个连接器之间，以防高速数字信号可能通过电场耦合或磁场耦合对输入输出环路产生差模干扰，并通过电缆发射出去。

所有的连接器最好都放置的印制板同一侧，因为一旦高速数字信号构成共模辐射源，作为共模辐射天线的电缆在两侧要比在一侧所产生的共模辐射大得多。

避免将连接器装在地层隔缝上，防止隔缝两边的地电位差产生共模干扰。

2. 地线和电源线的布置

地线与电源线的布置同样是关键，其原则如下：

① 区分不同地线：电源、数字电路与模拟电路、高速运算器件和低速器件、大电流与小电流信号电路要有各自分别的地线，只在适当的位置并联，以防止公共地线阻抗耦合（如图 10.13 所示的布置）。

② 双面板的数字地通常采用梳形结构和井字形网状结构，也不要让地线层隔缝阻挡高频回流的通路，因为大闭合面积的高频信号，会产生种种不良的后果：加大了与板上其他电路磁场耦合的可能性；增大了环路电感，容易使时钟电路等产生振荡；环路电感上的高频压降构成了共模辐射源，增加向空间的辐射干扰，同时易受空间磁场干扰。

③ 电源线和地线的布线应当结合起来考虑。

3. 信号线的布置

① 不相容的信号线应当相互隔离，尽量远离，避免平行。这里的不相容包括数字电路与模拟电路、高频信号与低频信号、大电流信号与小电流信号等。高速信号线特别是时钟

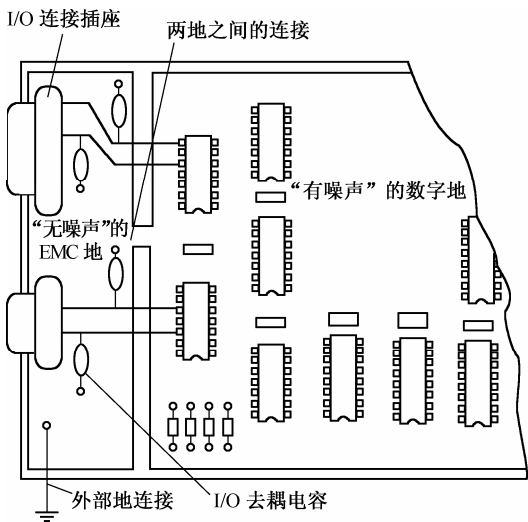


图 10.13 地线的布置

信号线要尽量短。信号线的布置最好根据信号的流向顺序布置。

② I/O 信号线和地线之间，在连接器端口处应加高频去耦电容，以抑制进出两个方向的差模干扰。

③ 尽量减小信号环路的面积，避免形成天线效应。做到板上的各部分功能电路都能够正常工作，没有相互干扰；尽可能降低对外的传导发射和辐射干扰，达到相关的标准要求；外部传导干扰和辐射干扰对印制板上的电路基本无影响。

④ 任何空闲的器件引脚和金属都应当有良好的接地，不允许有任何电气上没有连接并且悬空的金属存在。

10.3 电磁干扰的抑制、屏蔽与接地

电磁兼容 (Electromagnetic Compatibility, EMC) 是指，“设备（分系统、系统）在共同的电磁环境中能一起执行各自功能的共存状态，即该设备不会在受到同一电磁环境中其他设备的电磁发射时导致失效或遭受不允许的降级，它也不会使同一电磁环境中的其他设备（分系统、系统）因受其电磁发射而导致失效或遭受不允许的降级”。可见，电磁兼容研究的主要目的是，如何使在同一电磁环境下的各种电气子系统、设备、元件都能够正常工作，互不干扰，达到兼容状态。

微型计算机控制系统设计中，电磁兼容有着越来越重要的意义。这是因为，随着电子技术与电气自动化技术的飞速发展，电子设备和电气设备所产生的电磁污染日益严重。这些电磁污染反过来又严重影响了电子设备和电气设施的安全与可靠运行。电磁兼容作为一门新兴的综合性学科，在 20 世纪 60 年代诞生后得到了迅速的发展。所有的现代化领域，包括航空航天、军工、电力、通信、交通、计算机、医疗卫生与生命学科等都必须解决电磁兼容问题。近年来随着电力电子器件的不断发展，以及计算机技术在工业自动化检测控制中的广泛应用，计算机控制系统往往是由电力系统、功率电子设备、计算机及微电子装置互相结合形成强电和弱电配合工作的一体化系统，如何实现强电和弱电的电磁兼容性问题，是电磁兼容研究的主要问题之一，也是计算机控制系统可靠运行的一个关键性技术问题。

电磁干扰是影响系统稳定与可靠工作的主要因素之一。在某种程度上，也可以说电磁兼容研究的就是电磁干扰 (Electromagnetic Interference, EMI) 和抗干扰问题。电磁兼容研究的主要内容就是构成电磁干扰的三要素：电磁干扰源、干扰的传播途径和对干扰敏感的设备。包括干扰源特性研究，如干扰产生的机理、频域与时域的特性、表征其特性的主要参数、抑制其发射强度的方法等；敏感设备的抗干扰性能和保护措施；电磁干扰的传播特性与机理；电磁兼容数据的测量，如测量的设备、方法、检测数据的处理技术及测量结果的评价等；系统内与系统间的电磁兼容性；电磁干扰的危害等。

为了尽量减小客观存在的电磁干扰对检测控制信号与过程的影响，基本方法是抑制干扰源和保护被干扰对象。必须针对系统工作的电磁环境采取相应的处理措施，保障系统的安全可靠运行。

10.3.1 共模与串模干扰的抑制

设备与计算机之间的连接、计算机与计算机之间的互连等，导致共模干扰与串模干扰普遍存在于微机控制系统中。对于不同地电平之间的共模电压及模拟信号对地漏阻抗所导致的

共模干扰，主要的抑制措施是隔离，包括变压器隔离、光电隔离和浮地屏蔽技术。

1. 变压器隔离

变压器可以实现输入信号与输出信号之间的隔离，但是变压器两端的电路应分别采用两组互相独立的电源。若被隔离信号是高频信号，就可以直接进行隔离；如果被隔离信号是直流或者是低频信号，则须先通过调制电路变换成高频交流信号连接至变压器原边，再将变压器的副边输出信号通过解调器恢复成直流或低频信号，以某种形式送给计算机（参见图 10.11）。可以将调制器、变压器、解调器及放大电路做成一体，组成隔离放大器。这样把模拟信号电路与数字信号电路隔离，也就是把模拟地与数字地断开，使共模干扰电压不能构成回路，从而抑制了共模干扰。

2. 光电隔离

光电耦合器件的发光二极管在输入信号的控制下发光，传送给光敏三极管。光敏三极管的集电极和发射极作为光电耦合器的输出端，完成信号传输，如图10.14所示。

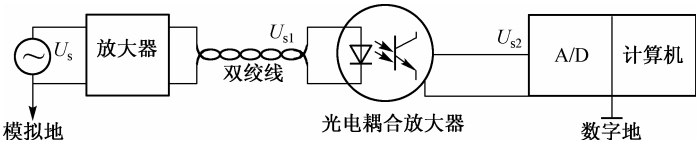


图 10.14 光电耦合器件的信号传输

光电耦合器的特点是：

- ① 由于密封在一个管壳内，不会受到外界光的干扰；
- ② 电—光—电的转换，切断了各部件之间电气互连，包括地线的联系；
- ③ 发光二极管动态电阻非常小，而干扰源的内阻一般很大，能够传送到光电耦合器输入端的干扰信号就变得很小；
- ④ 和晶体管的放大倍数相比，光电耦合器的传输比一般很小，远不如晶体管对干扰信号那样灵敏，而发光二极管只有在通过一定的电流时才能发光，即使干扰电压幅值较高，但由于没有足够的能量，仍然不能使其发光，从而可以有效地抑制掉干扰信号；
- ⑤ 无论干扰信号的幅值有多高，由于光敏三极管的驱动电压的限制，其输出端都不可能包含高电压尖峰干扰成分；
- ⑥ 光电耦合器提供了较好的带宽、较低的输入失调漂移和增益温度系数，能较好地满足控制信号传输速度的要求。

使用光电耦合器件隔离输入输出信号时，应注意：

- ① 两端的信号电路同样应由完全独立的电源供电；
- ② 若用于传输模拟信号，应当保证被传输信号的变化范围始终限制在光电耦合器件线性工作区内；同时，应当严格挑选光电耦合器件和采取相应的非线性校正措施，否则会产生较大的误差；
- ③ 与变压器隔离相比，光电耦合容易实现，成本低，体积小，效果好，大量应用于数字信号的隔离，被微机控制系统广泛采用。但与 TTL 门电路或 CPU 等高速器件相比，一般光电耦合器件的工作速度相对较低，不能直接与总线逻辑相连接，以免因为速度不匹配而无法正常工作。

双层屏蔽三线采样（ S_1, S_2, S_3 ）的浮地隔离式放大器如图 10.15(a)所示，这种方式提高了共

模输入阻抗，减小了共模电压在输入回路中引起的共模电流，从而具有较高的抗共模干扰能力，其等效电路如图10.15(b)所示。

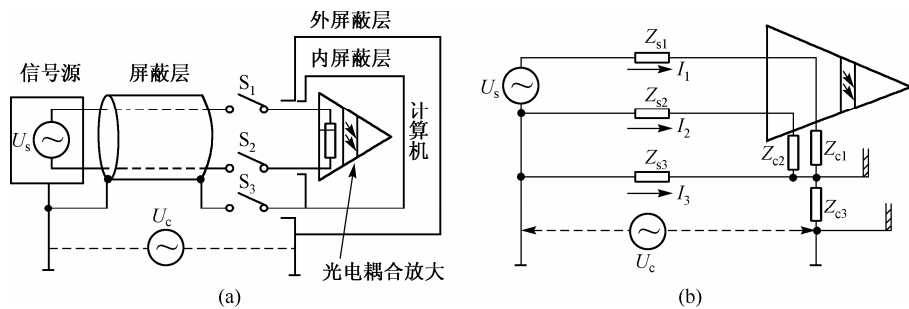


图 10.15 双层屏蔽三线采样电路及其等效电路

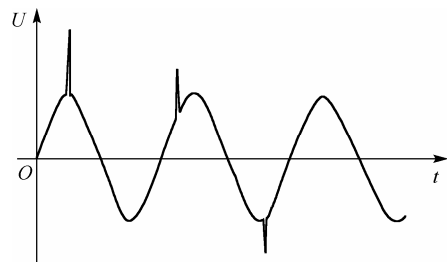


图 10.16 尖峰干扰示意图

3. 吸收与滤波

在有电流开关设备和变流设备的场合，尖峰干扰叠加在电源和输入信号上（如图10.16所示），是微机控制系统最主要和危害最严重的干扰源之一，甚至造成工业系统中器件和设备的击穿。一般采取远离干扰源，并联压敏电阻以吸收尖峰干扰，采用光电耦合电路和隔离变压器及“Watch-dog”技术等防止措施。

抑制串模干扰的最常用方法是滤波。在信号的输入端采用如图 10.17(a)所示的无源低通滤波器或如图 10.17(b)所示的有源低通滤波器，对于抑制串扰有较好的效果。根据串模干扰频率与被测信号频率的分布特性，决定选用具有低通、高通、带通等传递特性的滤波器。

无源滤波器由电阻、电容、电感等无源元件构成，缺点是信号有很大的衰减。

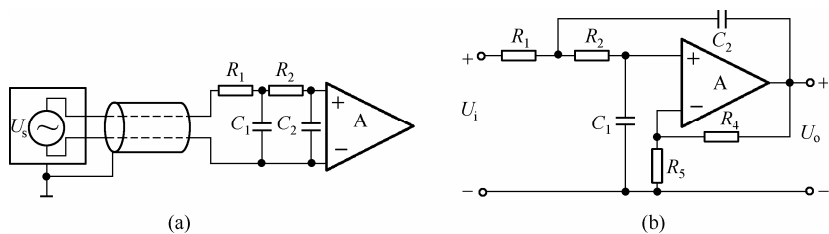


图 10.17 无源和有源低通滤波器

有源滤波器以反馈放大器为基础，不仅可提高信号增益，而且还具有较好的频率特性，特别是适用于弱信号的传输，缺点是线路复杂。

在实际的微机控制系统中，串模干扰的频率一般大于被测信号的频率，因此较多地使用低通滤波器，如无源阻容低通滤波器或有源低通滤波器。

差模和共模滤波器串联的电源滤波电路如图10.18所示，这种结构对交流与直流电源都适用。该电路的滤波性受两端负载阻抗的影响小，而且具有双向作用：不仅能够阻止电网中的噪声进入设备，也可以防止设备产生的干扰污染电网。在图 10.18 中， L_1 和 L_2 是两个单扼流圈，组成差模电感滤波器，电感量一般选几十 mH。 C_1 是差模电容滤波器，一般选 0.047~

0.22 μF 。 L_3 和 L_4 是共模扼流圈，绕在一个铁氧体环上，电感量约为几个 mH 。 C_2 和 C_3 是共模电容滤波器，电容量约为几个 nF 。若该值过大，将引起滤波器机壳漏电的危险。 C_2 和 C_3 的连接点是“接地”，即滤波器机壳，而按规定，机壳是接大地的。

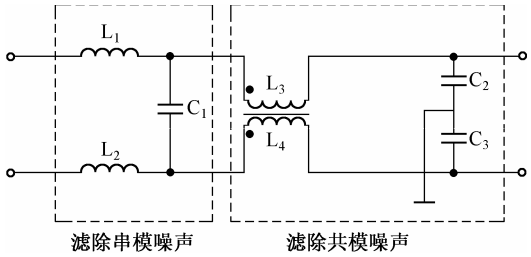


图 10.18 差模和共模滤波串联的电源滤波器

10.3.2 长线传输

为了避免长线传输产生地线环流、天线效应、波反射及信号衰减等问题，必须在接地、屏蔽及隔离等方面采取适当的措施。

在系统间长线传输的各个节点中，由于复杂的供电问题，各个节点之间地电平的不等，不同节点间的共模电压可能很高，常常会导致器件的击穿。因此，各个节点的传输信号必须采取变压器隔离或光电隔离等措施，同时必须分别独立供电。

采用终端阻抗匹配或始端阻抗匹配，可以消除长线传输中的波反射或者把它抑制到最低限度，并且减小信号的衰减。

传输线波阻抗 R_p 的测量如图 10.19 所示。调节可变电阻 R ，并用示波器观察门 A 的波形，当达到完全匹配时，即 $R = R_p$ 时，门 A 输出的波形不畸变，反射波完全消失，这时的 R 值就是该传输线的波阻抗 R_p 。

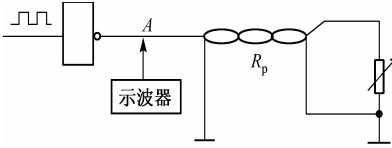


图 10.19 传输线波阻抗 R_p 的测量

根据传输线的基本理论，无损耗导线的波阻抗 R_p 为

$$R_p = \sqrt{\frac{L_0}{C_0}} \quad (10.2)$$

式中， L_0 为单位长度的等效电感 (H)； C_0 为单位长度的等效电容 (F)。

最简单的终端匹配方法如图 10.20(a) 所示。如果传输线的波阻抗是 R_p ，当 $R = R_p$ 时实现终端匹配，消除了波反射。此时，终端波形和始端波形的形状一致，只是时间上滞后。若终端电阻 R 变低，则增加了输出负载，波形的高电平下降，降低了高电平的抗干扰能力，但对波形的低电平没有影响。

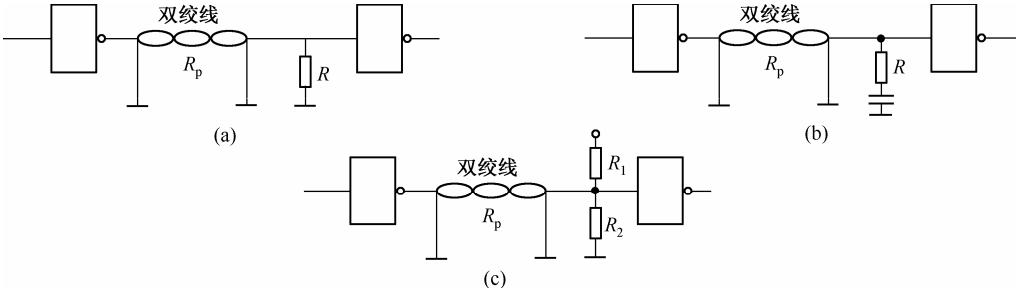


图 10.20 终端匹配方法

图 10.20(c) 所示的终端匹配方法可以克服上述缺点，其等效电阻 R 为

$$R = \frac{R_1 \times R_2}{R_1 + R_2} \quad (10.3)$$

适当调整 R_1 和 R_2 的阻值, 可使 $R = R_p$ 。这种消除波反射匹配方法的优点是波形的高电平下降较少, 缺点是低电平抬高, 从而降低了低电平的抗干扰能力。为了同时兼顾高电平和低电平两种情况, 可选取 $R_1 = R_2 = R_p$, 此时等效电阻 $R = R_p$ 。实践中常通过适当选取电阻 R_1 和 R_2 , 使 $R_1 > R_2$, 让低电平抬高较少而高电平降低稍多, 等效电阻仍然是 $R = R_p$ 。

在传输线始端串入电阻 R , 如图10.21所示, 也能基本上消除反射, 达到改善波形的目的。一般选择始端匹配电阻 R 为

$$R = R_p - R_{sc} \quad (10.4)$$

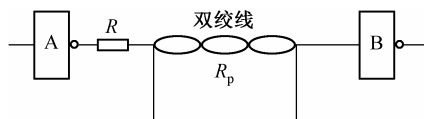


图 10.21 始端匹配

式中, R_{sc} 为门 A 输出低电平的输出阻抗。

这种匹配方法的优点是波形的高电平不变, 缺点是波形低电平会提高, 其原因是由终端门 B 的输入电流 I_{sr} 在始端匹配电阻 R 上的压降所造成的。显然, 终端所带负载门个数越多, 低电平抬高得越显著。

10.3.3 信号线的选择与布线

选择合适的信号线, 并正确地设计布线方式, 可以较好地抑制干扰, 特别是线间耦合干扰。

1. 信号线的选择

信号线的选择, 一般考虑抗干扰、经济和实用(施工方便)这三个方面。微机控制系统常用的信号线包括光缆、屏蔽和非屏蔽双绞线及同轴电缆等。

光缆不产生电磁干扰, 抗干扰能力也最强, 但是只用于传输数字信号, 而且施工比较困难, 在运动系统中要解决抗冲击与振动问题, 一般应用在严重干扰场合的信号传输。

使用铜线如双绞线和同轴电缆的领域较多。要求比较高, 或者干扰现象比较严重的现场, 采用屏蔽双绞线和同轴电缆是提高抗干扰能力的可行途径。单位长度绞数越多的双绞线抗干扰能力越好。要求同轴电缆有良好的屏蔽, 目前通常使用双屏蔽层的同轴电缆。

同轴电缆和双绞线(屏蔽双绞线与非屏蔽双绞线)在不同的接地方式时对磁场屏蔽效果的测试如图 10.22 所示。其中, 图 10.22(g)是同轴电缆在单端接地时的情况, 由于不受地环流的影响, 故屏蔽效果最好。由于双绞线对电场耦合没有屏蔽作用, 故图10.22(d)和(h)在没有地环流的条件下充分发挥了它对磁场的屏蔽作用。增加了屏蔽层的双绞线, 是中低频磁场屏蔽电路的优选方案。如图10.22(i)所示。图10.22(k)结合了(i)和(g)的优点, 因此效果比(i)还要好一些, 但一般不建议使用, 因为万一由于某种原因屏蔽层染上干扰噪声, 就可能流入信号线, 因此屏蔽层与信号线还是一点连接为好。

选择屏蔽电缆在使用时要注意屏蔽层的接地问题, 尤其是长线传输过程中的地线环流所引起的效应。在工业过程中, 由于长线连接的两个系统之间的地电平差, 一方面可能产生差模干扰, 另一方面可能有过大的地环流电流, 这两点都可能干扰测控系统的正常工作。在实际使用时, 可采用隔离变压器、光电耦合器等切除地环流。

不论是单端还是双端的屏蔽层接地, 都能够对电场产生屏蔽。在低频场合, 一般可采用屏蔽双绞线, 电路信号采用单端接地, 屏蔽层也采用单端接地, 有较好的屏蔽效果; 在高频场合, 最好采用同轴电缆, 信号单端接地, 屏蔽层也单端接地, 有非常好的屏蔽效果。

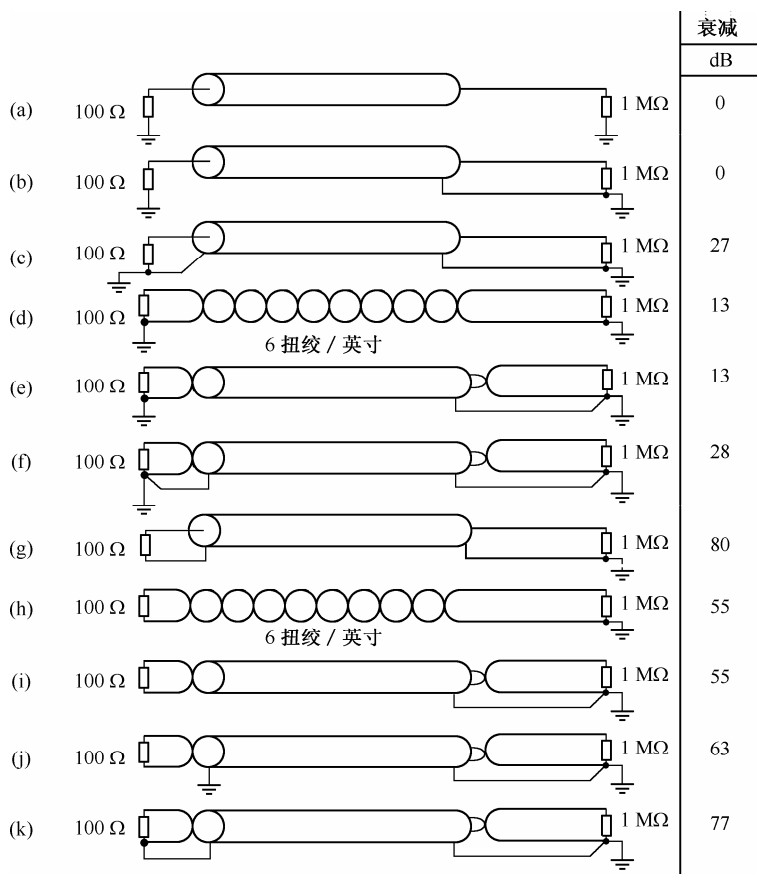


图 10.22 各种信号电缆在不同接地和屏蔽方式下对磁场的衰减

信号线的截面积一般采用 1.5 mm^2 和 1.0 mm^2 两种。多股线电缆的优点是可挠性好，适宜于电缆沟有拐角和狭窄的地方。

2. 信号线的布线

尽管在实际布线时很难做到远离干扰源，但通过合适的布线措施，使信号与电力线垂直或者尽量远离干扰源，可以抑制耦合干扰；否则，将会引入干扰。

信号线的布线要注意以下事项。

- ① 模拟信号线与数字信号线不能合用同一股电缆；
- ② 屏蔽信号线的屏蔽层要一端接地，同时要避免多点接地；
- ③ 信号线的走线要尽量远离干扰源，如避免敷设在大容量变压器、电动机等电器设备的近旁；信号电缆与电源电缆分开，将信号线单独穿管配线，并尽量防止平行敷设；

④ 如果由于现场条件有限，信号电缆与电源电缆不得不敷设在一起时，则应满足以下条件：在电缆沟内应当从上到下依次架设信号电缆、直流电源电缆、交流低压电缆、交流高压电缆，信号电缆与电源电缆之间设置与大地连接的金属隔板，或电源电缆使用屏蔽罩，或大于表 10.1 列出的最小间距。

表 10.1 信号线和电力线的最小间距

电力线容量		信号线和电力线 的最小间距 /cm
电压 /V	电流 /A	
125	10	12
250	50	18
440	200	24
5000	800	≥48

10.3.4 屏蔽与接地

1. 屏蔽技术

大电压小电流的噪声源，辐射形式主要表现为电场。低电压大电流的噪声源主要表现为磁场。如果噪声的波长和两者的距离满足条件 $d > \lambda / 2\pi$ 时，则噪声源的辐射表现为电磁场。

在实际系统中，电路中的电荷都是流动的，产生的电场也是交变的。良好的接地屏蔽体包围发射源或者被干扰体，可以完全消除交变电场的影响。

当磁场频率比较低时（100 kHz 以下），通常采用铁磁性材料，如铁、硅钢片、坡莫合金等，进行磁场屏蔽。为了取得比较好的效果，一般采用多层屏蔽叠加的屏蔽方式。

高频磁场的屏蔽材料采用金属良导体，如铜、铝等。由于高频电流的集肤效应，涡流只在屏蔽体表面的薄层中流过，所以在屏蔽高频磁场时屏蔽体不需要采用较厚的材料。在实际使用时屏蔽体应当接地，这样比较安全，又可以屏蔽电场。

电磁场屏蔽用于抑制噪声源，防止电磁场耦合产生的干扰对敏感设备的影响。电磁场屏蔽必须同时具有电场屏蔽和磁场屏蔽的效能，屏蔽材料采用电阻率较小的良导体。空间电磁波在入射到屏蔽体表面时屏蔽体表面会产生反射和吸收，良导体对低频电场的反射耗损较大，但对低频磁场的反射耗损较小。金属体越厚，吸收耗损越大。铁磁性材料的吸收耗损大于铜、铝等良导体。电磁波能量因反射和吸收被大大耗损，从而起到屏蔽的作用。

2. 接地

在微机控制系统设计中，正确采用接地和屏蔽措施，可以消除大部分的噪声干扰。

接地技术的作用主要是，消除各电路电流流经一个公共地线阻抗时所产生的噪声电压；避免受磁场和地电位差的影响，避免形成地环路及噪声耦合；使屏蔽和滤波形成回路；保障系统安全。

接地的含义是以一个等电位点或等电位面作为电路系统的基准电位。保护地线必须在大地的电位上。信号地线可以依据设计要求，采用大地电位或其他参考电位。

(1) 低频接地

在实际的微机控制系统中，输入输出通道的信号频率绝大部分处于 1 MHz 以下的低频段。由于地电平的不完全一致，信号地线的接地方式应采用一点接地，而不采用多点接地。

低频接地一般可采用串并联一点接地的接法。将符合噪声标准的各子系统经一组公共地线接地，也就是说，可用分组接法，即低电平电路、高电平电路、数字信号、模拟信号、大功率信号等各自的地线内部汇总后并联一点接地（如图 10.23 所示）。

(2) 通道馈线接地

通道的馈线必须一点接地。一个实际的模拟量信号输入通道总是可以简化为信号源、输入通道和放大器等几部分。若各个部分分别接地，则可能导致磁场耦合、地线环流与噪声干扰，因此应当采用单端接地的方式。如果必须采用多端接地方式，而又要排除地环流的噪声干扰，则应采用变压器隔离（适合中高频模拟信号）、隔离放大器（适合中低频或直流模拟信号）或者光电耦合器件（特别适合数字信号）将地环路隔断。另外，输入电缆线的屏蔽层和信号源公共端都应在信号源处接地。

(3) 安全接地

由于机箱或机柜内的强电设备可能通过感应或绝缘“漏电”，为了保证设备和人身的安全，应将机柜与内壳经短接后成为良好的导体，然后与大地可靠连接。

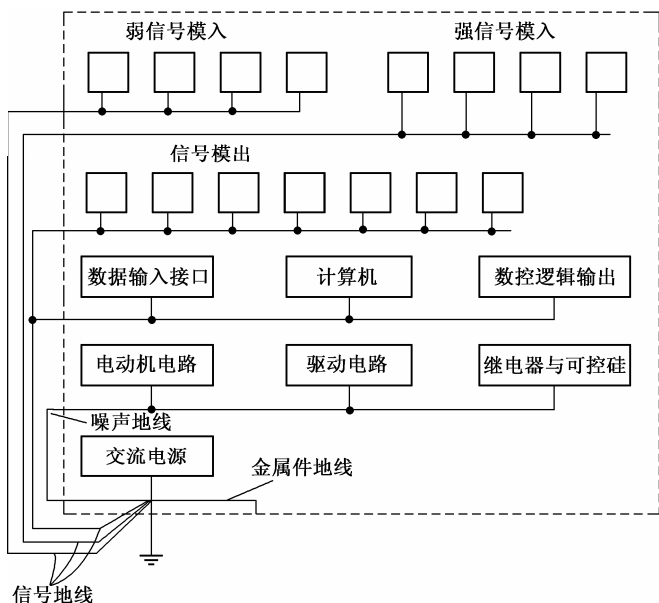


图 10.23 实用的一点接地

10.3.5 电源

微机控制系统一般都是直接或间接由交流电网供电，电网频率的波动、电压的大幅度涨落（浪涌）（工业电网的欠压或过压常常达到额定电压的 $\pm 15\%$ 以上，这种状况的持续时间可能长达几分钟、几小时甚至几天）、叠加的高达几百伏甚至几千伏的尖峰脉冲干扰、高次谐波和供电中断，都将直接影响到微机控制系统工作的可靠性和稳定性，必须采取电源保护措施。

1. 微机控制系统和大功率的动力负载分开供电

大功率用电设备的接通和断开、供电线路的切换操作、晶闸管变流器的工作，都会在供电电源上产生很严重的浪涌、谐波、高频噪声等。微机控制系统的供电应当有专门的供电设施，至少是有专门的电缆和专门的有良好接地的变压器，最好是安装线路滤波器。

2. 微机控制系统自身稳压电源的滤波与抗干扰性能

交流供电电源中包含了部分高频干扰信号，应当在稳压电源中串接低通滤波器，以滤除高频干扰信号，再向系统的各个部分提供电力。

现在的直流稳压电源一般都是开关电源，通过调节开关管的工作频率，以脉宽调制的方式来调整输出电压的幅值。优点是体积小、自身功耗低、效率高，特别是对输入电压的适应范围大，也具有相当好的抗干扰能力，但开关电源自身也可能产生高频干扰，必须有良好的滤波与噪声抑制措施。

3. 不间断电源（UPS）

UPS 在正常情况下是一个交流稳压器，当外界电源中断时，由 UPS 的后备电池通过逆变器向微机控制系统提供应急电源。UPS 包括功率半导体组成的变流稳压装置、线路切换装置和后备电池等。选用时应注意后备电池的性能和容量，逆变器的特性、保护功能、输出波形、切换时间及其谐波成分等。

4. 掉电检测

电源掉电检测的目的是，检测交流电源掉电故障，以保护计算机的运行现场，防止故障进一步扩大。掉电检测电路的输出一般连接至控制计算机的不可屏蔽中断。在运行过程中一旦发生掉电故障，能够及时发现并依靠电源装置的剩余能量完成运行现场的保护和重要数据的存储。恢复供电后，微机控制系统可以从断点处继续运行。

10.4 人机界面与监控组态软件

10.4.1 人机界面

1. 微机控制系统工作模式

由于被控对象、工作环境、在分布式测控网络中的节点位置及技术定位等的不同，微机控制系统有多种工作模式。

(1) 黑模式

系统不设置人机接口及显示装置。此模式通常用于分布式网络系统中的下位机等，直接面向控制对象，通常由该节点的上位机或智能终端实现管理，不需要人工干预，结构成本低，但是对可靠性的要求较高。

(2) 智能终端方式

系统没有专门的信息输出显示，在调试阶段可以通过 RS-232 或网络接口等与智能终端或监视器相连。此方式一般应用在网络系统的下位机、仿真开发装置等。

(3) 简易人机接口方式

通过简易的人机接口装置完成信息的显示及简单的命令输入。此方式通常用于独立的测控装置、智能仪器仪表、分布式测控网络系统的智能节点等。成本低、应用方便。

(4) 友好的人机接口方式。

一般有图像化的彩色显示、各种方便友好的命令输入装置，例如，各种按钮、开关、操纵拉杆等。

(5) 模拟屏方式

应用在需要各种模拟显示的特殊场合，采用大面积发光二极管显示屏、液晶投影装置等。

(6) 多人机接口方式

有多个显示输出装置和多种命令输入设施，用于仿真驾驶训练装置、大型自动检测监视系统等。

方式(1)、(2)和(3)结构简单，成本低廉，但往往是独立工作，因此对可靠性、故障自检能力及容错能力等要求较高。方式(4)多用于直接数字控制系统，如单回路调节器。方式(5)、(6)两种方式的成本较高。

2. 微机控制系统的人机界面

人机界面要根据成本、研制周期、应用环境及技术特征等因素来选取，但都必须考虑标准化与兼容性、可靠性与环境适应性、系统的技术特点，特别是人机工程学和工程心理学的要求。新技术的引入必须让人使用更加舒适，运行更加安全与可靠。

微机控制系统的人机界面主要包括信息的显示和命令与参数的输入两大部分。

(1) 信息显示技术

显示有各种形式, 包括图形输出, 如控制过程的变化曲线、各种参数显示等, 也可能是一些各种设备状态的显示, 如指示灯或更复杂的显示。

常用的显示器包括发光二极管 (Light-Emitting Diode, LED)、阴极射线管 (Cathode Ray Tube, CRT)、等离子体显示 (Plasma Display, PDP)、液晶显示 (Liquid Crystal Display, LCD)、场致发光 (Electroluminescent, EL)、有机电致发光 (Organic Light-Emitting Device, OLED) 及指示灯等。

① 发光二极管 (LED)。LED 是注入电致发光技术的一种: 半导体的 PN 结在加正偏压时产生少数载流子注入, 与多数载流子复合而发光, 因此又称为结型电致发光。发光二极管可以制成数码管或各类特定形状的符号显示器, 应用在各种数字显示和状态显示, 也可以制成点阵式单色或彩色通用图形显示器, 但单元尺寸和驱动功耗较大。

② 阴极射线管 (CRT)。CRT 是目前应用最普及的显示器, 其原理是由阴极发射的电子束经过加速、聚焦, 经偏转扫描电路控制的电子束击打荧光屏上的荧光粉层, 使得荧光粉层激发发光。彩色显示器则是通过电子束击打相应的 RGB 彩色荧光粉合成彩色。CRT 的特点是亮度高、色彩鲜艳、生产技术成熟、性价比高, 其缺点是体积太大、功耗大、抗振动和冲击能力差, 而且有一定的闪烁和静电等不良影响, 容易引起视觉疲劳。一般适合在宽敞及较固定的场合使用。

③ 液晶 (LCD)。液晶显示无闪烁, 也适宜于移动、有振动与冲击环境, 在航空航天、军事装备、工业和家用电视等各个领域得到了广泛应用。液晶是介于液体与晶体之间的第三种状态, 其外观是流动的混浊液体, 液晶显示的基本原理是利用液晶分子在电压、电流或电场的作用下其分子发生旋转、扭曲、重排或偏转的光学各向异性和双折射现象。液晶显示器由点阵驱动电路 (例如, TFT 薄膜三极管驱动电路)、液晶基体、透明电极、彩色偏振滤光、背光源、玻璃基板及接口电路等组成。优点是厚度薄、功耗低、无闪烁 (TFT 驱动)、无静电、生产工艺成熟。缺点是对工作温度范围、视角等有一定的限制。

④ 等离子体显示 (PDP)。PDP 在现有平板显示器中技术水平和视觉效果最高, 其优点是固有的存储特性、高亮度、高对比度、长寿命、大视角、真正的无扭曲平面化、色彩艳丽和工作温度范围宽等。由于其工作过程中不需要电磁场的电子束运动, 因此外界的电磁场不会对其产生干扰, 有极好的环境适应性, 特别适合在各种条件恶劣的场合。PDP 分为单基板式 (表面放电式) 交流 PDP、双基板式 (对向放电式) PDP 和脉冲存储直流驱动 PDP 三种。交流驱动 PDP 的基本原理是惰性气体中冷阴极辉光放电, 利用气体电离时产生的紫外线, 激励彩色荧光粉发光而实现彩色 PDP 显示, 其标称寿命一般为 2.5~3 万小时。

⑤ 场致发光 (EL)。EL 是一种高场发光技术, 其原理是荧光粉中的电子或由电极注入的电子在外加电场作用下, 在晶体内部加速, 碰撞发光中心, 使发光中心达到激发或离化态, 当电子回到基态时辐射发光。目前, 已有一些简单的彩色显示器投放市场。

⑥ 有机电致发光 (OLED)。某些有机材料能在电场作用下发光, 并且光电转换效率极高, 能快速点亮和熄灭, 亮度高, 工作电压低。类似于 LCD 驱动技术, OLED 也有由行列地址选中的单元被点亮的主动驱动与在 TFT (薄膜三极管阵列) 驱动下点亮的被动驱动之分。有 6 种彩色实现方式可采用, 例如, 三基色材料、采用三色滤色镜、采用激光共振等。除了 LCD 的优点之外, OLED 主动发光, 不需要背光, 因而更省电, 而且对比度高、色彩逼真、没有视角限制, 省电, 体积轻薄, 制造工艺简单。缺点是使用寿命问题目前还没有彻底解决。

(2) 命令与参数的输入

输入装置设计要考虑可靠性和实用。可以采用标准的计算机键盘，也可以是各类按钮、开关、旋钮、薄膜键、触摸屏等。近年来还出现了越来越多的人性化设计，如声控、光控输入装置。

10.4.2 监控组态软件的特点

组态软件作为用户可定制功能的软件平台工具，随着微机控制技术、网络技术和分布式控制系统的日趋成熟，也日益普及与推广。作为自动化系统中“水平”和“垂直”集成的桥梁和纽带，监控组态软件可以消除“自动化孤岛”，提高自动化工程的工作效率，减少系统的维护和升级费用。对于自动化技术由工业应用积极向楼宇自动化、家庭自动化、农业自动化、环境监测保护自动化等领域渗透，加快嵌入式系统的发展和信息化的步伐，起到了积极的促进作用。

组态(Configuration)的概念是，使用软件工具对计算机及软件的各种资源进行配置，使之自动执行特定任务，满足使用要求。监控组态软件是面向监控与数据采集(Supervisory Control And Data Acquisition, SCADA)的软件平台工具。随着它的快速发展，其任务已由最初的主要解决人机图形界面问题，发展为包括 SCADA、实时数据库、实时控制、通信及联网、开放数据接口、对 I/O 设备的广泛支持，等等。

国际比较知名的监控组态软件有 LabView, iFIX, Intouch, ONSPEC, LABTECH 等，不少国产组态软件也有较大影响。

监控组态软件有下列特点。

1. 实时多任务

监控组态软件最突出的特点是实时多任务。例如，数据采集与输出、数据处理与算法实现、图形显示及人机对话、实时数据的存储、检索管理、实时通信等多项任务要在同一台计算机上同时运行。

2. 可靠性高

当目标应用系统占用系统资源不超负荷时，监控组态软件应使最终生成的应用系统稳定可靠地运行。

3. 使用方便

监控组态软件的主要目的就是让使用者不需要修改软件程序的源代码，就可以生成适合自己需要的应用系统，灵活地解决下列问题：

- ① 与采集、控制设备间进行数据交换；
- ② 使来自设备的数据与计算机图形画面上的各元素关联起来；
- ③ 处理数据报警及系统报警；
- ④ 存储历史数据并支持历史数据的查询；
- ⑤ 各类报表的生成和打印输出；
- ⑥ 提供灵活多变的组态工具，可以适应不同应用领域的需求；
- ⑦ 具有与第三方程序的接口，方便数据共享。

在组态软件中，使用者只需填写一些事先设计的表格，再利用图形功能把被控对象（如反应罐、温度计、锅炉、趋势曲线、报表等）形象地画出来，通过内部数据连接把被控对象的属性与 I/O 设备的实时数据进行逻辑连接。由组态软件生成的应用系统投入运行后，与被控对象相连的 I/O 设备数据发生的变化，会直接带动被控对象的属性变化，还可以方便地对

应用系统进行修改。

10.4.3 组态软件的数据流与典型构成

组态软件的典型构成包括图形界面系统，实时数据库系统，I/O 设备驱动程序，通信程序组件和应用程序管理器。

1. 组态软件的数据流

如图10.24所示，组态软件通过 I/O 驱动程序从现场设备获得实时数据，在进行必要的加工后，一方面以图形方式直观地显示在计算机屏幕上，并且当发生报警时及时以声音、图像的方式通知操作人员，另一方面按照组态要求和操作人员的指令，将控制数据回送给 I/O 设备，对执行机构实施控制或调整控制参数。

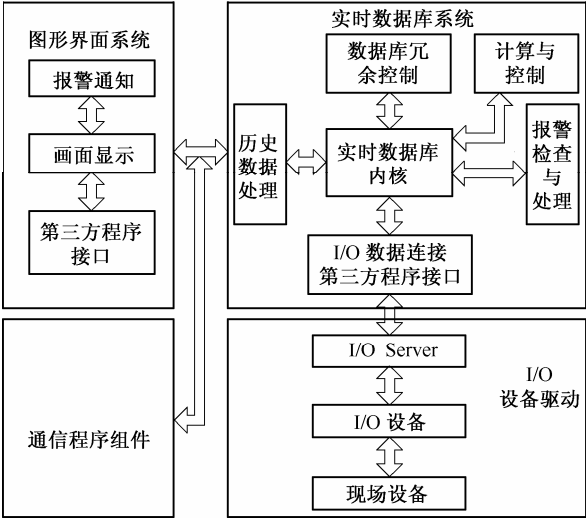


图 10.24 组态软件的数据处理流程

实时数据库是组态软件的核心和引擎，历史数据的存储与检索、报警处理与存储、数据的运算处理、数据库冗余控制、I/O 数据连接都是由实时数据库系统完成的。图形界面系统、I/O 驱动程序等组件以实时数据库为核心，通过高效的内部协议相互通信，共享数据。

2. 图形界面系统

图形界面系统包括图形界面开发程序和图形界面运行程序。

图形界面开发程序是使用者为实施控制方案，在图形编辑工具的支持下，进行图形系统生成工作所依赖的开发环境。它依照操作系统的图形标准，采用面向对象的图形技术，为用户提供丰富强大的绘图编辑、动画连接和脚本工具，帮助使用者简化操作。

所有的操作画面，包括流程画面都是在图形开发环境下编辑制作的，通过建立一系列用户数据文件，生成最终的图形目标应用系统，供图形运行环境运行时使用。

为了生成图形画面，构成现场各过程图形的画面被划分成一些简单的对象，例如，线、填充形状和文本。每个简单的对象均有影响其外观的属性。对象的基本属性包括线的颜色、填充颜色、高度、宽度、取向、位置移动等。这些属性可以是静态的，也可以是动态的。

静态属性在系统投入运行后保持不变，与组态时初始情况一致；动态属性则随来自 I/O 设备的变量，或由变量和运算符组成的数学表达式值的变化实时地改变。

图形对象的动画属性可以根据数据变量或表达式变化。例如，用一个矩形填充体模拟现场的液位，在组态这个矩形的填充属性时，指定代表液位的工位号名称、液位的上下限及对应的填充高度，就完成了液位的图形组态。建立画面中，图形对象与数据变量或表达式的对应关系的组态过程，通常称为动画连接。

在图形界面上，还具备报警通知及确认、报表组态及打印、历史数据查询与显示等功能。各种报警、报表、趋势曲线都是动画连接的对象，其数据源都可以通过组态来指定。这样每个画面的内容就可以根据实际情况由工程技术人员灵活设计，每幅画面中的对象数量均不受限制。

在图形界面中，各类组态软件普遍提供一种称为脚本语言的编程工具来扩充其功能，用脚本语言编写的程序段与对象密切相关，可由事件驱动或周期性地执行。例如，可以指定当按下某个按钮时，执行一段脚本语言程序，完成特定的控制功能，也可以指定当某一变量的值变化到关键值以下时，启动一段脚本语言程序，完成相应的控制功能。

在系统运行环境下，图形目标应用系统被图形界面运行程序装入计算机内存，并投入实时运行。

3. 实时数据库

现代工程和时间关键型应用，既需要数据库来支持大量数据的共享，维护其数据的一致性，又需要实时处理来支持其任务（事务）与数据的定时限制。一方面要求维护大量共享数据和控制数据，保证它们的完整性和一致性，尤其是时间一致性；另一方面又要求在规定的时刻及一定的时间内完成其处理，所处理的数据也有一定的时效性，过时则有新的数据产生，而当前的决策或推导变成无效。

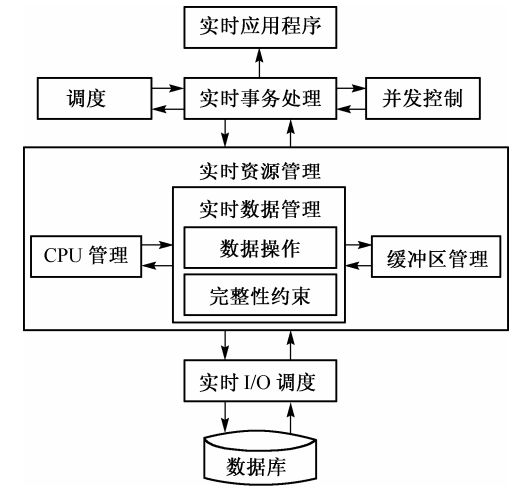


图 10.25 实时数据库的体系结构

实时数据库就是其数据和事务都有显式定时限制的数据库。系统的正确性不仅依赖于事务的逻辑结果，而且依赖于该逻辑结果所产生的时间。在数据库技术和实时系统“无缝集成”（Seamless Integration）基础上发展起来的实时数据库同时支持定时性和一致性。在数据库的结构与组织，数据处理的优先级控制、调度和并发控制协议与算法，数据和事务特性的语义及其与一致性、正确性的关系，数据查询/事务处理算法与优化，I/O 调度、恢复、通信的协议与算法等概念、理论、技术、方法和机制方面，都有着自身的特点（如图10.25 所示）。

(1) 实时数据库的特征

① 实时数据库的时间相关性特征

主要表现在数据与时间相关和事务的定时限制上。

- 数据与时间相关。按照与之相关的时间性质的不同，又可分为两类。时间本身就是数据，即从“时间域”中取值。如“数据采集时间”属于“用户定义的时间”，系统将毫无区别地把它像其他数据一样处理。随时间变化的数据值只在一定的时间内有意义，过时则无效。数据库中的数据是对其所面向的“客观世界”中对象状态的描述，

若对象状态发生变化,则引起数据库中相应数据值的变化,因而与数据值变化关联的时间可以是现实对象状态变化的事件发生的实际时间,称为“真实”或“事件”时间,也可以是数据库中相应数据值变化的时间,称为“事务时间”(任何对数据库的操作都必须通过一个事务进行)。实时数据的导出数据也是实时数据,与之关联的时间自然是事务时间。

- 实时事务的定时限制。“截止时间”是典型的定时限制。对于实时数据库,结果产生的时间与结果本身一样重要,一般只允许事务存取“当前有效”的数据。事务必须维护数据库中数据的“事件一致性”,而外部环境的反应时间要求也对事务施以定时限制。

② 实时数据库的数据特征

在实时数据库中,数据随外部环境状态的变化而快速变化,其值只在一定的时间内有意义,过时则无效。故系统除了维护数据库内部状态(数据值)的正确性和相容性外,还必须同时维护内部状态与外部环境实际状态的一致性,以及数据用来决策或推导新数据时在时间上的相互一致性。

实时数据库中,一个数据对象 d 由当前值 dv 、采样时刻 dtp 、外部有效期 $devi$ (外部现实对象状态变化的时间间隔)等 3 个分量组成,数据特征包括内部一致性、外部一致性和相互一致性。

内部一致性是指, dv 满足预先定义的数据库内部状态的完整性和一致性限制。这就是传统意义下的数据正确性。

外部一致性是指,设 tc 为当前或检测时间,当且仅当 $(tc-dtp) \leq devi$ 时, dv 和对应的外部现实对象的状态是一样的。

相互一致性是指,用来决策或导出新数据的一组相关数据称为一个相互一致集,记为 R ,其中的数据必须尽可能地在一个允许的公共时间期内被取出(或导出),这个公共时间期就称为 R 的相互有效期,记为 $Rmvi$,对于 R 中的任两个数据 d 和 d' ,有 $\{dtp-d'tp\} \leq Rmvi$,则说 R 中的数据是相互一致的。

具备了内部一致性、外部一致性和相互一致性的数据才是正确的。

③ 实时数据库的事务特征

实时任务往往有内部结构及其相互之间的联系。在实时数据库事务表现的特征中,最重要的是其定时性特征。

定时可以是绝对时间、相对时间或周期时间。实时数据库的定时性一方面来自数据的时间一致性,典型形式是周期或定期性限制,如“每 5s 取样一次”、“7:00 打印报表”等。定时性的另一方面来自系统对外界变化反应时间的要求,典型形式是施加于非周期事务的截止时间限制,如“若温度达到 1000℃,则在 1s 内发出报警信息”。

定时性包含两方面的含义:定时限制,即事务的执行有显式的时限,如指定的开始时间、截止时间等,它要求实时数据库必须有时间处理机制;定时正确性,即事务能按指定的时间要求正确执行,它要求权衡定时限制与数据一致性要求等多方面因素,提供合适的调度与并发控制算法。

(2) 实时数据库系统的功能

组态软件的实时数据库功能具体表现在数据处理功能、并发处理功能、在线组态查询功

能、对外开放功能等方面（如图10.26所示）。实时数据库的数据库管理系统既具有一般 DBMS 的基本功能，又有其特有功能。但首先是满足对事务的定时限制，其基本原则是，宁可要部分正确而及时的信息，也不要绝对正确但过时的信息。系统性能指标是满足定时限制的事务的比率，并确保硬实时事务的截止期，必要时甚至宁可牺牲数据的准确性与一致性。

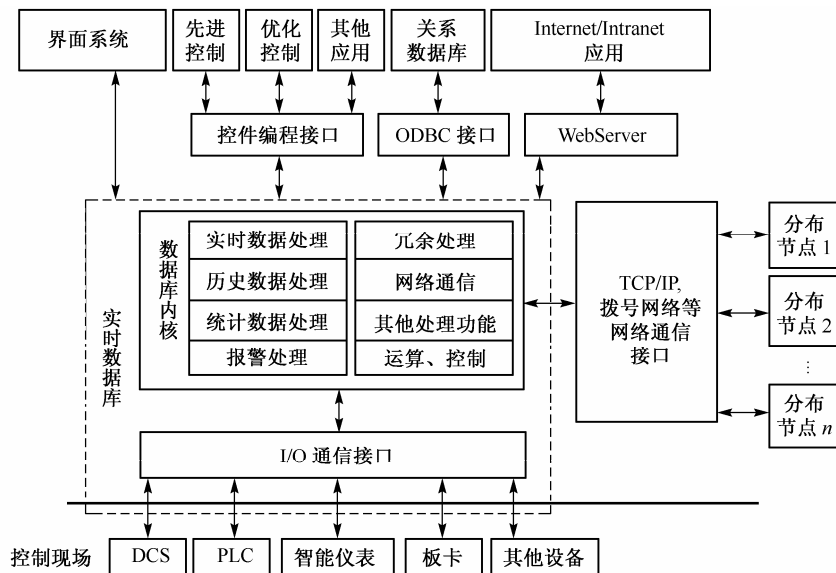


图 10.26 实时数据库系统的功能示意图

① 一般 DBMS 的基本功能

永久数据管理，包括数据库的定义、存储、维护等；有效的数据存取、各种数据操作、查询处理、存取方法、完整性检查；事务管理，事务的概念、调度与并发控制、执行管理；存取控制，安全性检验；对数据库的可靠性进行控制。

② 实时数据库的特有功能

尽可能地保持数据库的状态是当前外界最真实状态的映像；确保事务读取的数据是时间一致的；确保事务的及时处理，使其定时限制尤其是执行的截止期得以满足。

支持多种 I/O 设备的数据采集与回送，支持多种接口和数据交换方式，可同时连接多台不同类型的 I/O 设备，对所连接的所有各种类型的 I/O 设备及同一个设备上不同的 I/O 点参数都可以指定不同的扫描周期。

③ 实时事务的模型与处理

实时数据必须使用复杂事务模型，即嵌套、分裂/合并、合作和通信等事务模型。事务之间有多种交互行动和同步，存在结构、数据、行为、时间上的相关性及在执行方面的相互依赖性。

实时数据库中的事务有多种定时限制，其中最典型的是事务截止期。系统必须能让截止期更早或更紧急的事务较早地执行，也就是能控制事务的执行顺序。所以，需要根据截止期和紧迫度来标明事务的优先级，然后按优先级进行事务调度。

对于实时数据库事务，允许“放松的可串行化”或“暂缓可串行化”并发控制。

4. I/O 设备驱动程序

设备驱动程序（I/O Server）是组态软件与现场设备交互通信的桥梁，各种测控设备越来

越多地采用标准通信接口，使互联通信越来越简便。

(1) 设备驱动程序的数据流

如图10.27所示，I/O Server 直接负责从设备采集实时数据并将操作命令下达给设备，其可靠性直接影响着组态软件的性能。I/O Server 按照接口协议的规定向设备发送数据请求命令，对返回数据进行拆包，从中分离出所需数据（即组态的数据连接项和设备状态数据）。使用者只需组态数据库变量与设备数据项的对应关系。I/O Server 将组态的数据连接项根据接口协议的要求分类，把通过一次读写操作能够处理的数据连接项存放为一个数据包。数据处理以数据包为单位进行，大大提高通信效率。

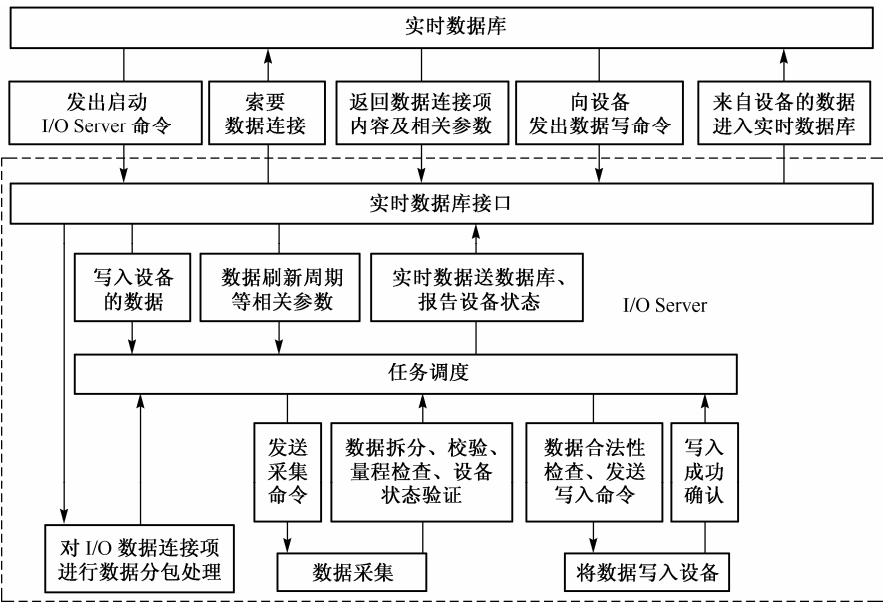


图 10.27 设备驱动程序的数据流

(2) 设备驱动程序的主要功能

① 进行链路维护

I/O Server 要将来自设备的数据转换成实时数据库需要的数据类型（实数、整型数、字符或字符串型等），同时对越界数据进行合理解释。

为提高数据的采集效率和数据安全，I/O Server 将所有数据连接项划分成只读、只写、可读写 3 种类型，以防止误写、误读的发生。

在同时处理多个同类设备时，对于故障设备，I/O Server 要减少访问频次，以免影响整个系统的数据刷新周期。当故障设备恢复正常后，再将设备的采集周期恢复原值。

② 管理输出队列

针对设备的特殊要求，I/O Server 对来自操作员的 I/O 命令逐级进行合法性检查，如首先检查是否只读数据，再检查数值是否越界等。

③ 与实时数据库系统进行无缝连接

I/O Server 与实时数据库间通常采用高效的内部通信方式，如进程间通信、直接内存映射、OLE 方式等，可以迅速报告设备的故障信息，产生系统报警。对来自设备的数据进行初步预处理，在一段时间内数值没有发生变化的数据可以不向实时数据库传送或减少传送次数，以

进一步提高数据通信的效率。

(3) 设备驱动程序的使用要领

① 正确设置 I/O Server 的采样周期。因为采样周期太快，可能引起设备死机或工作不正常；太慢又会影响数据刷新速度。

② 界面系统的回写数据周期不要小于采样周期的 3 倍。因为数据回写任务的优先级较高，频繁回写将影响数据采集。

③ 及时处理系统报警，设置安全密码，防止非法操作。

5. 通信程序组件和应用程序管理器

应用程序管理器使应用程序的搜索、备份、解压缩、建立新应用等操作变得非常简单。

通信程序组件是开放系统的标志，也是组态软件与第三方程序交互及实现远程数据访问的重要手段，其主要作用如下：

- ① 在双机冗余系统中，进行主机与从机间的通信；
- ② 在构建分布式 HMI/SCADA 时进行多机间的通信；
- ③ 在基于 Internet 或 Browser/Server (B/S) 应用中实现通信功能。

习题与思考题

- 1 微机控制系统的设计有哪些基本原则？请访问有经验的微机控制工程技术人员，总结他们对微机控制系统的设计原则有哪些充实和提高。
- 2 微机控制系统的设计过程包括几个阶段？每个阶段应注意什么关键问题？
- 3 微机控制系统中，信息的显示和命令与参数的输入分别有哪些主要方式，它们各有什么优缺点？
- 4 为了正确可靠地进行数字量（开关量）和模拟量的输入输出，设计输入输出模板应遵循哪些原则？可以采取哪些措施？
- 5 试结合课题，设计 I/V 转换和 F/V 转换等常用的模拟量信号调整电路。
- 6 电磁兼容的主要研究内容是什么？为什么所有的现代化领域都必须解决电磁兼容问题？
- 7 分析电场耦合、磁场耦合与公共阻抗耦合产生干扰的机理。
- 8 微型计算机控制系统可能受到来自哪些方面的电磁干扰？可以采取哪些措施来抑制干扰？
- 9 在微机控制系统设计中，怎样正确采取接地和屏蔽措施来消除大部分的噪声干扰？
- 10 简述一个监控组态软件的应用实例，并总结其优缺点。

第 11 章 微机控制技术实验

实验是学习微型计算机控制技术的重要环节。本章以专门研发的 DP-01 多 MCU 实验系统为平台,围绕多种控制算法、数据传输方式与总线在智能家居中的应用,提供了设计性、综合性实验课题,以发挥学习者的想象力和创造性。大纲中给出的场景和流程意在抛砖引玉,“5+”要求可供进一步探讨的参考。

11.1 实验大纲

11.1.1 顺序控制及其在门禁系统中的应用

实验类别: 设计 实验学时: 2

基本要求:

- ① 掌握按规定时间顺序或事件顺序进行动作的顺序控制算法;
- ② 根据顺序控制算法编制应用程序;
- ③ 掌握开关量输入、输出的硬件电路;
- ④ 应用顺序控制算法,设计开关量控制的软硬件系统并实现预定功能。

主要设备与器材:

- ① PC 机;
- ② MCU 实验平台

预习要求:

熟悉 MCU 实验平台中下列模块的原理、硬件接线和软件模块,以便选用:

语音实验模块,独立控制的 LED、拨动开关、键盘模块,红外收发模块,非接触式 IC 卡模块或接触式 IC 卡模块,LED 点阵显示模块或 LCD 模块,蜂鸣器模块,继电器及其驱动模块。

基本内容:

- ① 设计一个密码锁;
- ② 设计门禁系统的模拟场景和 workflows;
- ③ 完成模拟门禁系统中各模块与 MCU 的硬件连接;
- ④ 应用各模块软件,编制、调试门禁系统的系统软件;
- ⑤ 实现门禁系统的预定功能。

参考场景:

- ① 用键盘模拟密码锁并实现其功能,例如:

将键盘输入的数字与 MCU 中存储的密码对比,若符合,则提示下一步操作——用语音实验模块播送“请放入 IC 卡”;若不符合,则播送“对不起,密码不正确”。

- ② 用非接触式 IC 卡模块或接触式 IC 卡模块模拟出入卡并实现其功能,例如:

将 IC 卡读出数据与 E²PROM 中存储的身份识别码对比,若符合,则发出身份验证通过信号(例如,用 LED 点阵显示模块或 LCD 模块显示欢迎字样);若不符合,则发出报警信号(例如,用语音实验模块发出警告)。

- ③ 用红外收发模块模拟防止非法入侵装置并实现其功能，例如：
当红外收发模块接收到反射回来的红外信号时，发出报警信号（点亮红色 LED）。
- ④ 用继电器及其驱动模块模拟门的开启与关闭，并实现门禁系统的预定功能，例如：
当密码锁、出入卡和防止非法入侵装置的动作符合设定逻辑关系时，继电器吸合。

参考流程图：如图 11.1 所示。

“5+”要求：

- ① 分析各模块的与/或逻辑关系，提出门禁子系统的多种可行方案；
- ② 探讨加强门禁子系统安全性能的措施，例如，提高密码锁抗攻击性，增加红外探头的联动性等。

11.1.2 数字滤波及其在灯光子系统中的应用

实验类别：设计 实验学时：2

基本要求：

- ① 掌握模拟量的有源低通滤波、A/D 转换的硬件电路和 A/D 转换软件；
- ② 掌握数字滤波概念并编制应用程序实现高通、低通和带通数字滤波的算法；
- ③ 应用模拟量输入电路，高通、低通和带通的数字滤波算法，设计模拟量控制的软硬件系统并实现预定功能。

主要设备与器材：

- ① PC 机；② MCU 实验平台

预习要求：

熟悉 MCU 实验平台中下列模块的原理、硬件接线和软件模块，以便选用：
蜂鸣器模块，运算放大器模块，电压基准源模块，串行模数转换模块，PWM 电压转换模块，LED 模块。

基本内容：

- ① 应用蜂鸣器模块，编制播放一曲电子音乐的程序并调试实现；
- ② 设计音频输入的有源低通滤波、A/D 转换的硬件电路和 A/D 转换软件，并实现其模拟量输入功能；
- ③ 编制、调试对输入音频模拟量进行高通、低通和带通的数字滤波程序；
- ④ 编制、调试利用输入音频的高、中、低频分量，转换为 PWM 输出，控制 R、G、B 各色 LED 亮度的程序；
- ⑤ 设计应用输入音频各频段分量分别控制 RGB 亮度的软硬件系统，并实现彩灯系统的预定功能。

参考场景：

- ① 应用蜂鸣器模块播放一曲电子音乐；
- ② 将蜂鸣器模块输出的音频信号用运算放大器模块进行有源低通滤波后，经 A/D 转换后输入 MCU；

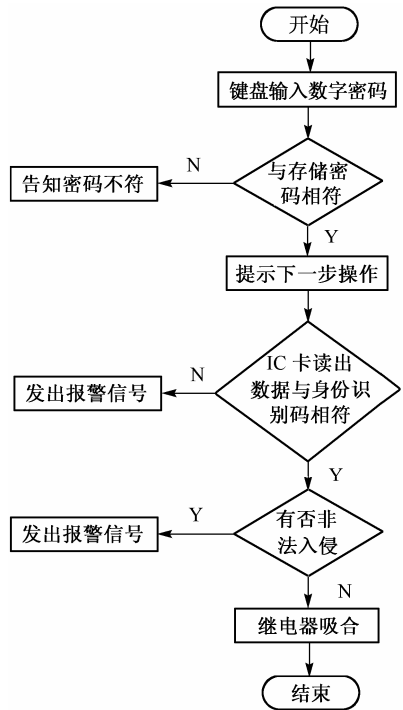


图 11.1 顺序控制及其在门禁系统的应用参考流程图

- ③ 对输入音频模拟量进行高通、低通和带通的数字滤波，将输入音频的高、中、低频分量变为 PWM 输出，送入 LED 彩灯模块控制 R, G, B 各色 LED 亮度；
 - ④ 在播放电子音乐的同时，LED 彩灯模块的彩色随音乐抑扬而变幻。
- 参考流程图：如图 11.2 所示。

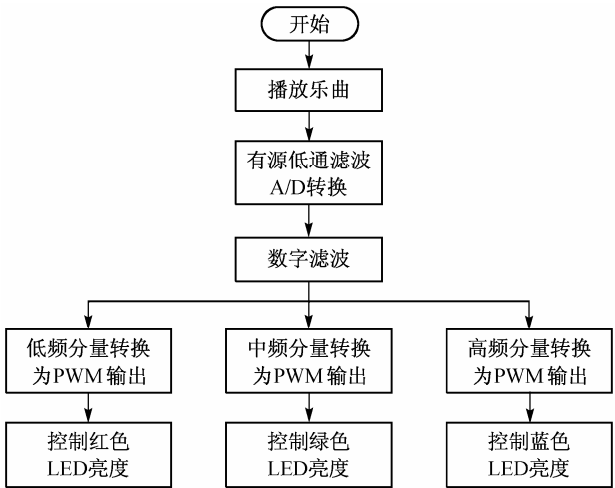


图 11.2 数字滤波及其在灯光子系统的应用参考流程图

“5+”要求：

- ① 对随音调变化不明显的基色采取加强变化的改进措施；
- ② 探讨对音频信号中特殊成分变化的处理方法；
- ③ 探讨 LED 彩灯的色彩可表达语言感情变化的“解语花”实现方案。

11.1.3 运动控制及其在输送子系统中的应用

实验类别：设计 实验学时：2

基本要求：

- ① 通过 PWM 输出，实现直流电动机调速功能；
- ② 通过 I/O 输出，实现步进电动机调速功能；
- ③ 应用直流电动机或步进电机构成输送子系统，实现预定控制功能。

主要设备与器材：

- ① PC 机；② MCU 实验平台

预习要求：

熟悉 MCU 实验平台中下列模块的原理、硬件接线和软件模块，以便选用：PWM 电压转换模块，直流电机模块，步进电机模块，红外收发模块。

基本内容：

- ① 应用 PWM 电压转换和直流电机模块实现直流电动机调速功能；
- ② 应用 I/O 输出和步进电机模块实现步进电动机调速功能；
- ③ 设计输送子系统的运行曲线，编制程序并调试实现。

参考场景：

- ① 设计输送子系统的运行曲线，给定启动、加速、保持、减速和停止各段坐标；
- ② 用直流电机或步进电机分别模拟输送子系统，按照控制要求，实现输送子系统运行功能。

参考流程图：如图 11.3 所示。

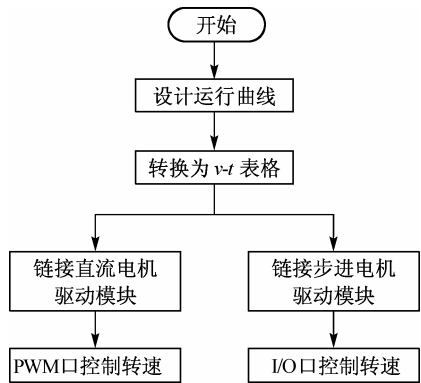


图 11.3 运动控制及其在输送子系统的应用参考流程图

“5+” 要求：

- ① 应用红外收发模块构成转速测量装置，检测直流电机和步进电机的转速；
- ② 比较直流电机模块和步进电机模块的调速控制功能，提出改进方法；
- ③ 探讨利用电动窗帘、灯光、音乐和语音实现“自然醒闹钟”的设计方案，并模拟预定功能。

11.1.4 PID控制及其在温度子系统中的应用

实验类别：设计 实验学时：2

基本要求：

- ① 掌握温度测量硬件电路和数据采集软件；
- ② 实现根据温度变化的 PID 增量控制，并调试实现温度子系统预定控制功能。

主要设备与器材：

- ① PC 机；② MCU 实验平台

预习要求：

熟悉 MCU 实验平台中下列模块的原理、硬件接线和软件模块，以便选用：数字温度采集模块，PWM 电压转换模块，蜂鸣器模块，语音实验模块。

基本内容：

- ① 利用数字温度采集模块实现温度测量和数据采集功能；
- ② 编制、调试实现增量 PID 控制功能的程序；
- ③ 设计并调试实现温度子系统预定控制功能。

参考场景：

- ① 对温度采集模块的升温装置通电、断电，采集不同状态下的温度测量数据；
- ② 利用 PWM 电压转换模块的输出电压，加热温度采集模块的升温装置，采集升温曲线数据；
- ③ 编制增量 PID 控制程序，使温度尽可能保持某恒定值，并将测量数据与单纯开关控制的测量数据相比较；
- ④ 实现温度检测的报警和语言提示。

参考流程图：如图 11.4 所示。

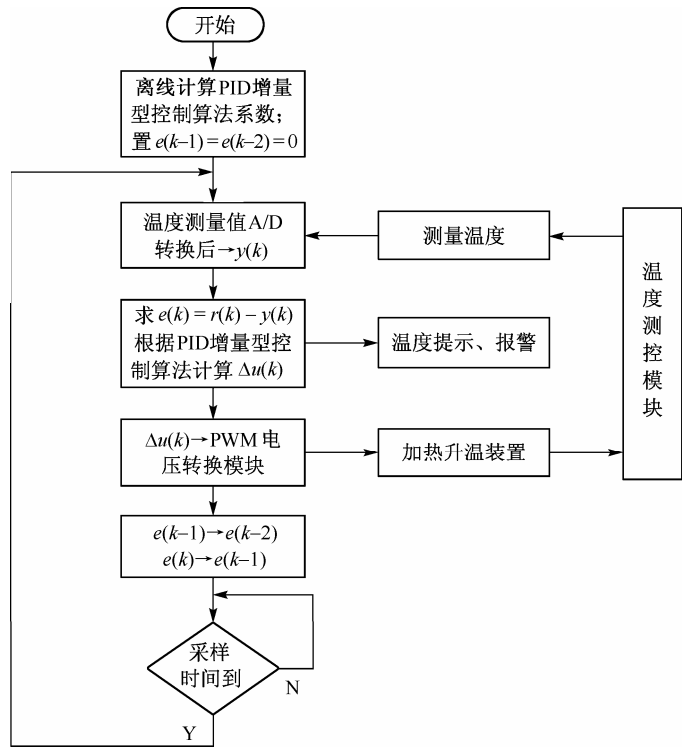


图 11.4 PID 控制及其在温度子系统的应用参考流程图

“5+”要求：

- ① 对滞后时间较长的升温装置，采集升温曲线数据，提出控制方案并检验其控制效果；
- ② 探讨提高温度控制精度和抗干扰能力的方案；
- ③ 探讨通过仿真调试寻求最佳参数的方案。

11.1.1.5 模糊控制及其在智能终端子系统中的应用

实验类别：设计 实验学时：2

基本要求：

- ① 掌握模糊控制器的设计，确定其输入变量和输出变量；
- ② 学会编制模糊控制规则表、调试模糊化、模糊决策和解模糊程序；
- ③ 应用模糊控制实现智能终端的控制功能。

主要设备与器材：

- ① PC 机；② MCU 实验平台

预习要求：

熟悉 MCU 实验平台中下列模块的原理、硬件接线和软件模块，以便选用：

数字温度采集模块，PWM 电压转换模块，继电器及其驱动模块，独立控制的 LED、拨动开关、键盘模块。

基本内容：

- ① 设计电饭煲模糊控制器，确定其输入变量和输出变量；
- ② 编制模糊控制规则表，调试模糊化、模糊决策和解模糊程序；

③ 设计、编制模糊控制程序，并仿真调试实现电饭煲的控制功能。

参考场景：

① 用拨动开关的组合模拟给定炊饭量 Q ，用数字温度采集模块模拟锅底温度传感器，用 PWM 电压转换模块加热升温装置模拟加热功率；

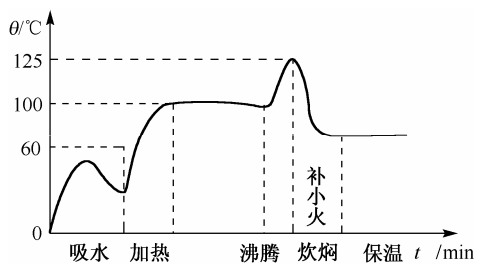


图 11.5 米饭炊制最佳加温曲线

② 用继电器模拟电饭煲的得电开关。上电后计时，以最佳加温曲线（如图11.5 所示）上的时间对应点为基准温度，根据锅底温度传感器检测出的信息，计算出实测温度与基准温度之差 E 和二者升温率之差 SE ，应用模糊推理给出加热功率 P 的控制决策。若温差 E 小，温升 SE 大，则降低功率；若温差 E 大，温升 SE 小，则应增加功率。逼近最佳加温曲线。

③ 加热控制模糊变量的模糊子集可规定为

基准温差 E ：负大、负中、负小、负零、正零、正小、正中、正大；

温差变化率 SE ：负大、负中、负小、零、正小、正中、正大；

功率控制 P ：0，PS，PM，PB。

参考流程图：如图 11.5 所示。

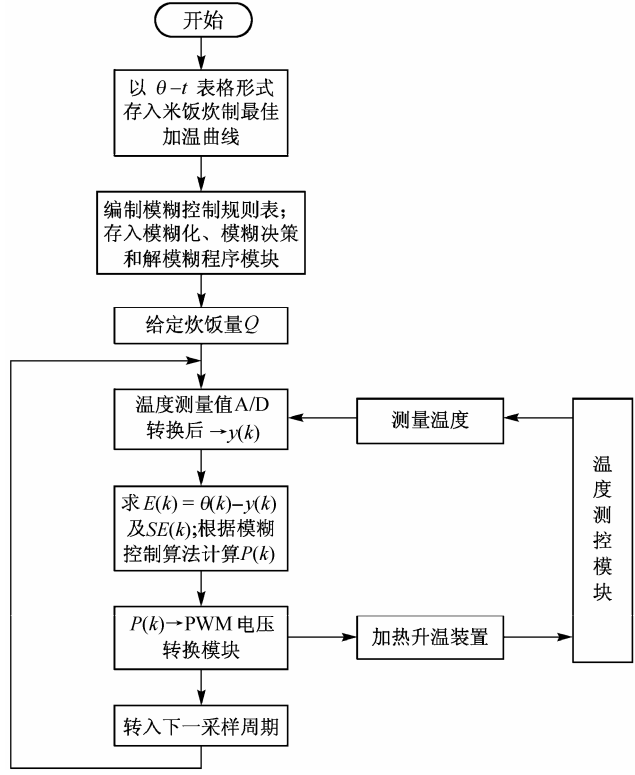


图 11.6 模糊控制及其在智能终端子系统中的应用参考流程图

“5+” 要求：

① 完善电饭煲的人性化措施，例如，定时、显示、提示功能；

② 探讨扩大电饭煲功能多样化的方案，例如，煮粥、煲汤。

11.1.6 多种数据传输方式与总线在智能家居中的应用

实验类别：综合设计 实验学时：2

基本要求：

应用多种接口方式进行数据传输，并用多种方式显示控制效果。

主要设备与器材（2组以上）：

① PC 机；② MCU 实验平台

预习要求：

熟悉 MCU 实验平台中下列模块的原理、硬件接线和软件模块，以便选用：

RS485 模块，USB 接口（USB1.1 接口或 USB2.0 接口）模块，TCP/IP 模块，红外收发模块，蜂鸣器模块，LED 点阵显示模块，继电器及其驱动模块，独立控制的 LED、拨动开关、键盘模块。

基本内容：

① 应用 RS485 总线、USB 接口（USB1.1 接口或 USB2.0 接口）、基于以太网接口的 TCP/IP 协议、应用红外基带和红外编码进行数据传输；

② 应用蜂鸣器、LED、继电器等多种方式显示控制效果。

参考场景：

用 2 组 MCU 实验平台采用多种接口进行数据收发，并分别用不同的方式显示收到数据后的控制效果。

“5+”要求：

① 用多组 MCU 实验平台构成数据传输链路，每一传输环节各采用一种接口，每台 MCU 实验平台各选用一种方式显示收到数据后的控制效果；

② 应用红外收发模块实现对空调、电视等家电的红外遥控；

③ 提出智能家居的其他数据传输方式并制订应用方案。

11.1.7 网络与各子系统的整合在智能家居中的应用

实验类别：综合设计 实验学时：4

基本要求：

组合各子系统并用总线互连，构成功能较完备的复杂系统。

主要设备与器材（多组）：

① PC 机；② MCU 实验平台

预习要求：

熟悉 MCU 实验平台中各模块的原理、硬件接线和软件模块：

完成前 6 个实验的基本要求，设计应用总线将各子系统整合为功能较完备的网络系统的方案。

基本内容：

① 设计智能家居的应用场景；

② 应用 MCU 实验平台的各个模块，模拟智能家居控制的各子系统并编制整合程序；

③ 调试网络系统，实现智能家居的模拟场景控制功能。

参考场景：

① 用电动机模块、蜂鸣器模块模拟自然醒闹钟，如下：

起床时间快到了，窗帘缓缓拉开，喜爱的乐曲由轻到响，引发人体的自然反应，在阳光和音乐的早晨，逐渐从深睡过渡到浅睡，再舒适地醒来；

② 用独立控制的 LED、拨动开关、键盘模块、红外收发模块和继电器及其驱动模块模拟家电的定时控制和红外遥控；

③ 用彩灯模块、蜂鸣器模块模拟伴着音乐翩翩起舞的彩灯。

“5+”要求：

① 完善自然醒闹钟功能，例如，灯光控制、免迟到报警；

② 提出智能家居走向未来生活的进一步设想，并设计实现方案。

11.2 DP-01 多MCU实验平台

DP-01 多 MCU 实验平台可选用 MCS-51，HC(S)08，PIC 等系列中多种 MCU 芯片，用 C 语言和汇编语言进行微控制器（单片机）原理与应用实验，配备了丰富的软硬件模块和外围器件设备，通过 MCU 总线接口可以根据需要灵活组合，进行微型计算机控制技术的设计性、综合性实验，发挥使用者的积极性、主动性和创新性。

11.2.1 DP-01 多MCU实验平台布局

DP-01 多 MCU 实验平台布局如图11.7所示。

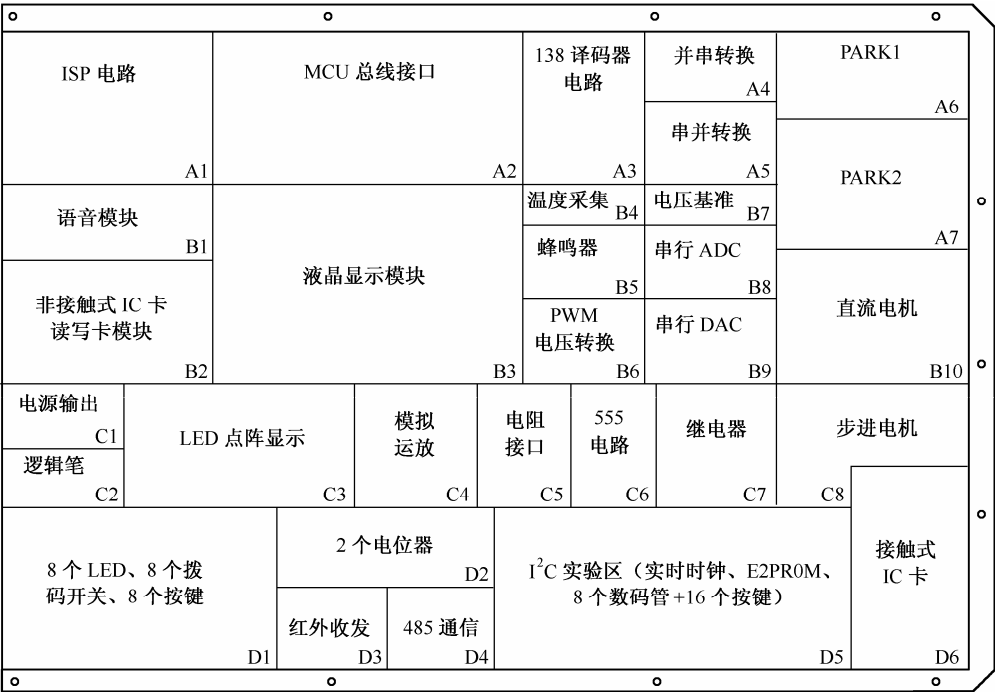


图 11.7 DP-01 多 MCU 实验平台布局

HC(S)08 系列 MCU 及其接口连接模块 E1 和 PIC 系列 MCU 及其接口连接模块 F1 可另外固定，与 MCU 总线接口连接，在图11.7中未画出。

11.2.2 DP-01 多MCU实验平台功能模块

DP-01 多 MCU 实验平台功能模块如表 11.1 所示。

表 11.1 DP-01 多 MCU 实验平台功能模块

编 号	模 块 名 称	器件与功能说明
A1	PHILIPS 系列 MCU（51 内核）	程序调试、下载、通信
E1	Freescale HC(S)08 系列 MCU	程序调试、下载、通信
F1	PIC 系列 MCU	程序调试、下载、通信
A2	总线 I/O 扩展	引出 MCU 的各功能引脚； 含 74HC573 锁存器
A3	3-8 译码	74LS138 译码器
A4	并转串	74LS165 并行-串行转换器
A5	串转并	74LS164 串行-并行转换器
A6	PARK1	连接各种扩展 PARK 模块，例如，USB 接口、CAN-bus 接口、以太网接口等
A7	PARK2	同 A6
B1	语音	ZLG1420A 语音模块、麦克风、扬声器
B2	读非接触式 IC 卡	ZLG500A 非接触式 IC 卡读卡模块与天线接口；时钟源与 8 路分频输出接口
B3	LCD 显示	128×64 图形点阵或 16×2 字符点阵
B4	温度传感器	18B20 单总线数字温度传感器
B5	蜂鸣器	交流蜂鸣器及其驱动电路
B6	PWM 输出	将 PWM 信号转换为电压输出
B7	电压基准源	TL431 电压基准源
B8	串行 A/D 转换器	TLC549 串行 8 位 A/D 转换器
B9	串行 D/A 转换器	TLC5620 串行 8 位 4 通道 D/A 转换器
B10	直流电机	直流电机及其驱动电路
C1	电源输出接口	+5 V、± 12 V 电源接口
C2	逻辑笔	通过 LED 显示 TTL 逻辑电平高低
C3	LED 点阵	16×16 LED 点阵及其驱动电路
C4	运算放大器	LM324 运算放大器
C5	电阻接口	为运算放大器提供电阻接口
C6	555 定时器	555 定时器及相关电阻、电容接口
C7	继电器	继电器及其驱动电路
C8	步进电机	步进电机及其驱动电路
D1	I/O 调试	独立的 LED 发光二极管、拨动开关、按键各 8 个
D2	可调电阻	10 kΩ和 1 kΩ可调电阻各 1 个
D3	红外收发	红外发射器和带解码红外接收器各 1 个
D4	RS485	SP485 用于 RS485 的电平驱动和接收
D5	I ² C	24WC02EEPROM（256 字节）、PCF8563 实时时钟及其外围电路、ZLG7290 键盘 LED 驱动芯片、8 段数码管 8 个、按键 16 个
D6	读接触式 IC 卡	读卡头可连接 SLE4442 卡

参 考 文 献

- [1] 于海生. 计算机控制技术. 北京: 机械工业出版社, 2007.
- [2] 周德泽. 计算机智能监测控制系统的设计及应用. 北京: 清华大学出版社, 2002.
- [3] 王晓明. 电动机的单片机控制. 北京: 北京航空航天大学出版社, 2002.
- [4] 桑楠. 嵌入式系统原理及应用开发技术. 北京: 北京航空航天大学出版社, 2002.
- [5] 周德泽. 计算机智能监测控制系统的设计及应用. 北京: 清华大学出版社, 2002.
- [6] 白英彩. 英汉计算机技术大辞典(新世纪版). 上海: 上海交通大学出版社, 2001.
- [7] 谢剑英. 微型计算机控制技术. 北京: 国防工业出版社, 2001.
- [8] 陈炳和. 计算机控制系统基础. 北京: 北京航空航天大学出版社, 2001.
- [9] 高金源. 计算机控制系统——理论、设计与实现. 北京: 北京航空航天大学出版社, 2001.
- [10] 王慧. 计算机控制系统. 北京: 化学工业出版社, 2000.
- [11] 齐蓉. 可编程控制器教程. 西安: 西北工业大学出版社, 2000.
- [12] 何玉彬. 神经网络控制技术及其应用. 北京: 科学出版社, 2000.
- [13] 蔡德聪. 工业控制计算机实时操作系统技术. 北京: 清华大学出版社, 1999.
- [14] 阳宪惠. 现场总线技术. 北京: 清华大学出版社, 1999.
- [15] 何克忠. 计算机控制系统. 北京: 清华大学出版社, 1998.
- [16] 袁南儿. 计算机新型控制策略及其应用. 北京: 清华大学出版社, 1998.
- [17] 蔡自兴. 智能控制——基础与应用. 北京: 国防工业出版社, 1998.
- [18] 张乃尧. 神经网络与模糊控制. 北京: 清华大学出版社, 1998.
- [19] 白英彩. 计算机集成系统——CIMS 概论. 北京: 清华大学出版社, 1997.
- [20] 吴启迪. 柔性制造自动化的原理与实践. 北京: 清华大学出版社, 1997.
- [21] 蔡自兴. 人工智能及其应用. 北京: 清华大学出版社, 1996.
- [22] 窦振中. 模糊逻辑控制技术及其应用. 北京: 北京航空航天大学出版社, 1995.
- [23] 涂时亮. 单片微机控制技术. 上海: 复旦大学出版社, 1994.
- [24] 王常力. 集散型控制系统的设计与应用. 北京: 清华大学出版社, 1993.
- [25] 陈汝全. 实用微机与单片机控制技术. 成都: 电子科技大学出版社, 1993.
- [26] 金以慧. 过程控制. 北京: 清华大学出版社, 1993.
- [27] 王锦标. 过程计算机控制. 北京: 清华大学出版社, 1992.
- [28] 严隽薇. 计算机实时控制软件设计导论. 北京: 清华大学出版社, 1990.
- [29] 胡寿松. 自动控制原理[M].第5版.北京: 科学出版社, 2007.
- [30] 潘新民, 王燕芳. 微型计算机控制技术. 北京: 电子工业出版社, 2003.
- [31] 吴坚, 赵英凯, 黄玉清. 计算机控制系统. 武汉: 武汉理工大学出版社, 2002.
- [32] Richard C. Dorf, Robert H. Bishop. 现代控制系统. 北京: 科学出版社, 2002.
- [33] Derenzo S.E.微机接口技术实验教程(英文版). 北京: 机械工业出版社, 2004.

- [34] Richard C. Dor, Robert H.Bishop. Modern Control Systems.Addison Wesley Longman, 2001.
- [35] Lefteri H.Tsoukalas ,et al. Fuzzy and Neural Approaches in Engineering. John wiley & Sons, 2001.
- [36] B.Wayne Bequette. Process Control: Modeling, Design, and Simulation. Prentice hall, 2003.
- [37] Roy E.Fraser. Process Measurement and Control. Prentice hall, 2001.
- [38] Jon Stenerson. Fundaments of Programmable Logic Controllers, Sensors and Communication. Prentice hall,1999.
- [39] 高木章二. デジタル制御入門. Ohmsha,1998.
- [40] 広井和男・宮田 朗. シミュレーションで学ぶ自動制御技術入門. CQ 出版社, 2005.